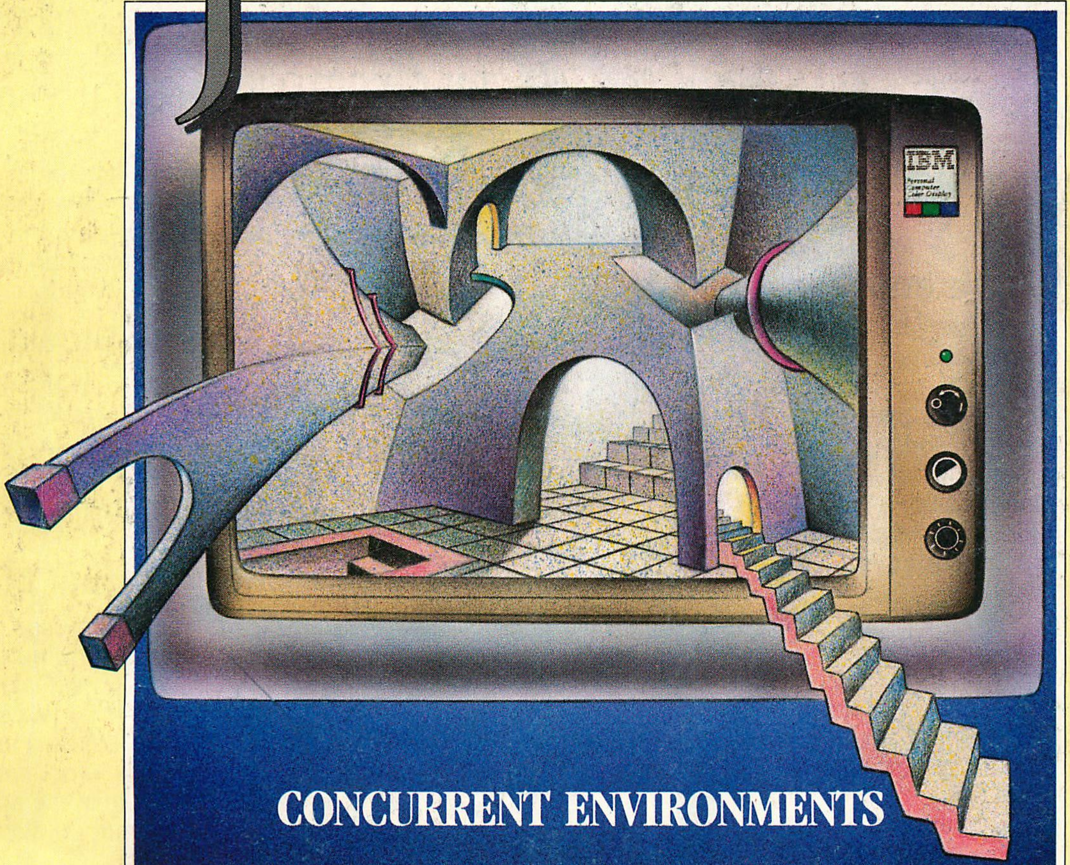
DECEMBER 1985 VOL. 3, No. 12 $3.95

FOR IBM PERSONAL COMPUTER USERS

PC TECH JOURNAL®



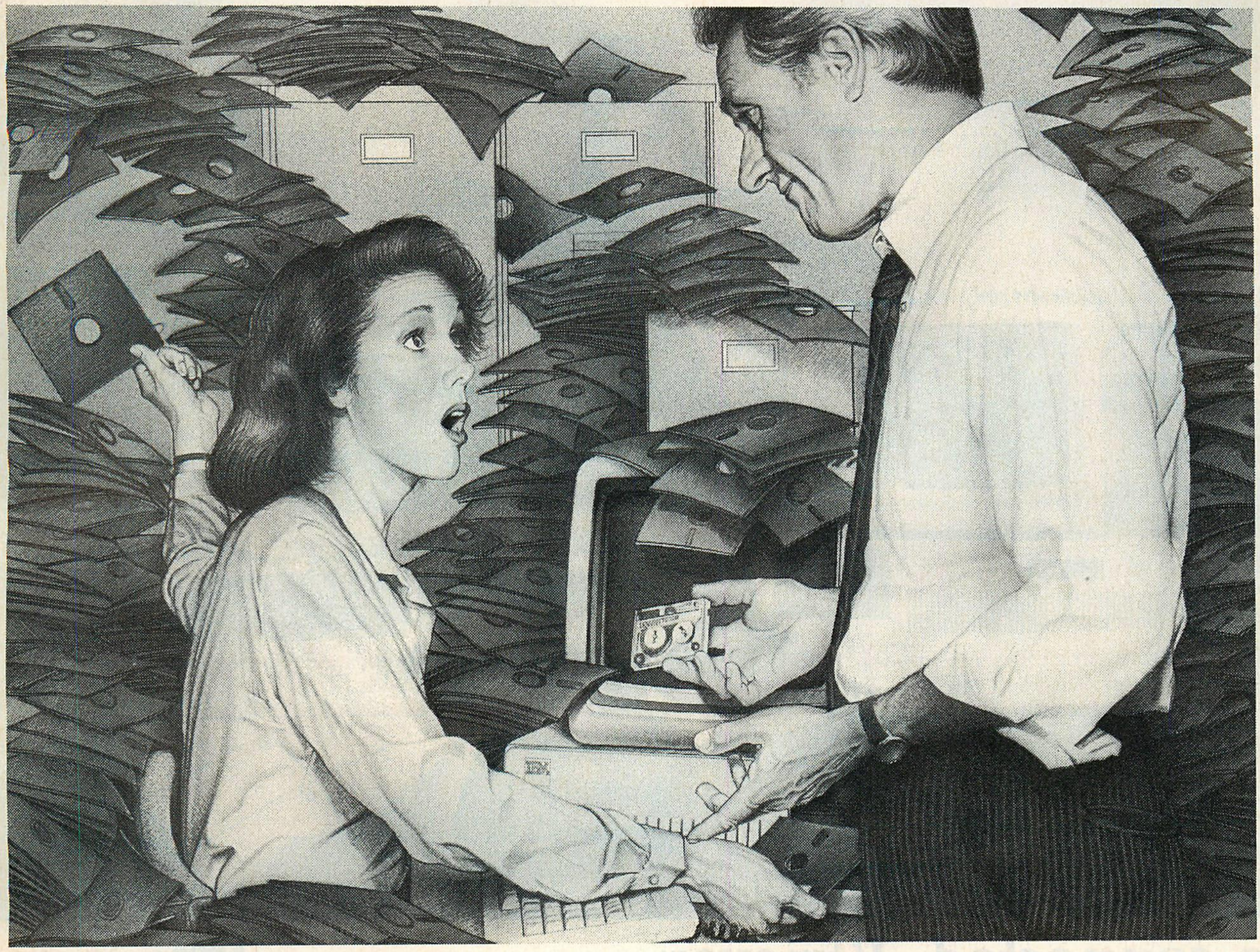
CONCURRENT ENVIRONMENTS

COPY-PROTECTION TECHNIQUES

PROLOG AND LISP

DATA MANAGER: INFORMIX-SQL

One Irwin BACKUP™ Tape can do what???



Add profit to each IBM PC/XT and AT sale!

With Irwin BACKUP™ Tape Systems, you add revenue with every IBM PC/XT, AT or compatible you sell. And you provide your customers convenience as well as protection from the unexpected. Irwin margins help you generate high profits on both ten and twenty megabyte BACKUP™ Systems. Each system can be mounted internally or set along side the computer in its own attractive case. The Irwin software, designed to operate on the IBM PC/XT and AT, is complete and very easy to install.

Your customers will easily understand the need for back up. When they see how easy it is to use an Irwin BACKUP™ Tape System and how little it costs, the only thing you'll have to do is write up the order.

And remember, you're selling more than a back up system. You're providing protection for your customers' accounts receivable, payroll, and other valuable data files, recorded on a tape cartridge so small that it fits easily in a shirt pocket out of harm's way.

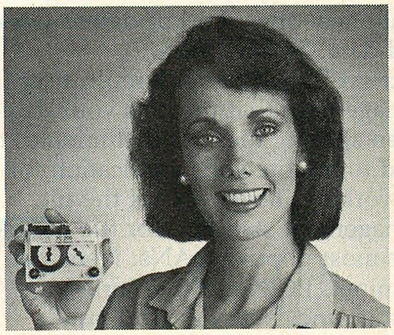
With almost one hundred thousand systems shipped, Irwin, the leader in back up systems for microcomputers, should be a part of your IBM PC/XT, AT computer line.

To get the complete story on how you can become an Irwin dealer, call toll free, **1-800/421-1879**.

CIRCLE NO. 228 ON READER SERVICE CARD

FAST, EASY, INEXPENSIVE BACKUP.

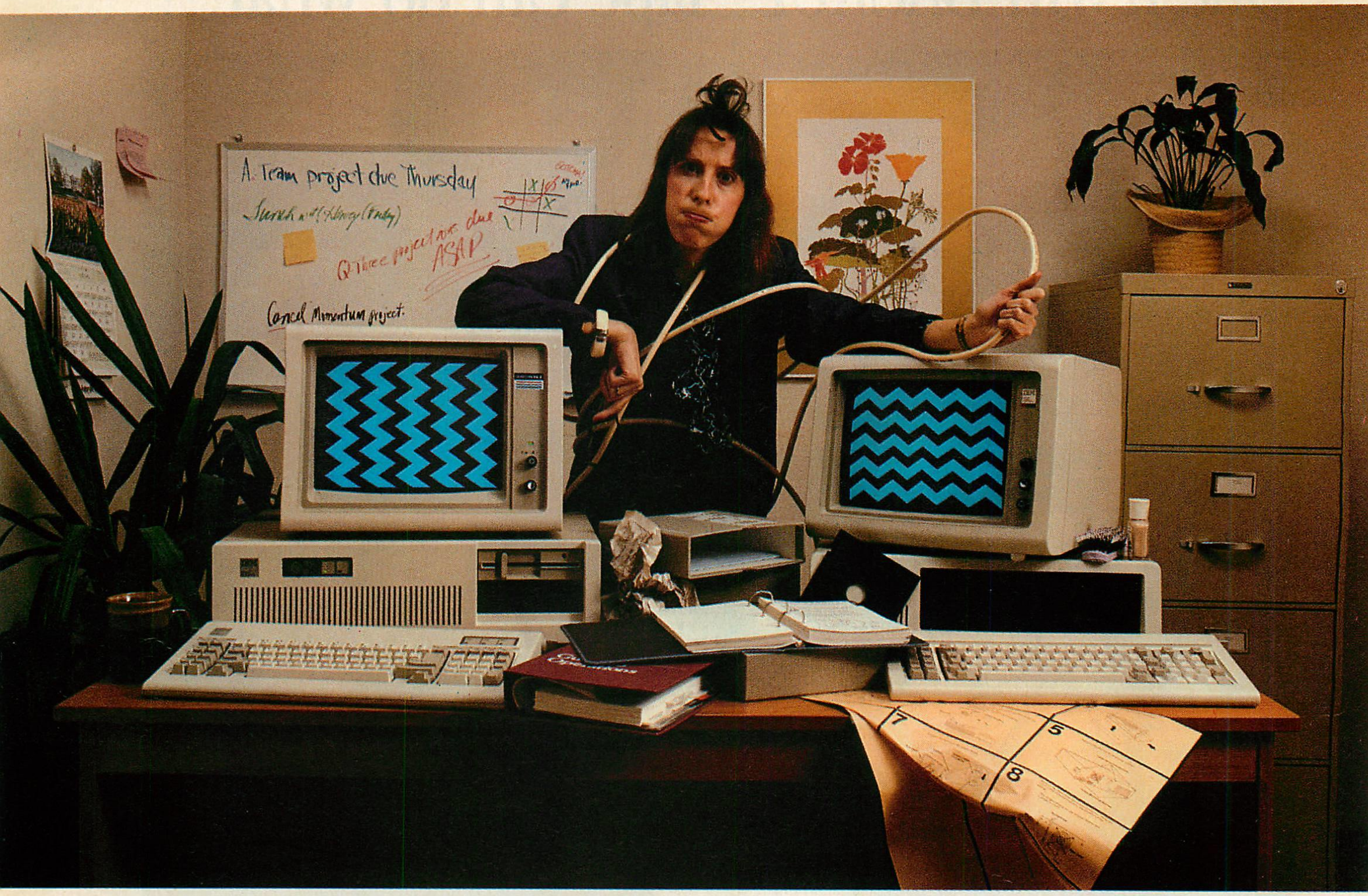
"No Problem, Thanks Irwin."



IRWIN™ MAGNETICS

2311 Green Road
Ann Arbor, MI 48105
(313) 996-3300 • 1-800/421-1879

If you can't share files on PC Network, you're using the wrong file manager.



Be connected. Btrieve.™

Networks can solve problems. But running a single-user file manager can create new ones: Lost updates. Garbled data. Trashed files.

Btrieve™/N offers *safe* multi-user file management that protects your data when sharing files. And eliminates the need to rewrite your application for networking. Btrieve/N set the file management standard for the industry's most popular LANs: Netware, Omninet, PC Net, EtherSeries, Nestar, and NetOne. And now Btrieve/N sets the standard for IBM's PC Network.

Fast. Btrieve/N is fast, too. It's written in assembly language especially for the IBM PC. And based on b-tree file indexing with automatic balancing for access speed that won't degrade as your database grows. With Btrieve/N, your applications always run fast. And users don't waste time waiting.

Automatic file recovery. Btrieve/N provides automatic file recovery after a system crash, so accidents and power failures don't turn into database disasters. Your Btrieve data always comes back intact.

Fully-relational data management. Using SoftCraft's entire family of products gives you a complete, fully-relational database management system. Rtrieve™/N adds report writing capabilities for generating the reports you need. Xtrieve™/N speeds users through database queries with interactive, on-screen menus—no command language or special syntax.

For professional programmers. Btrieve/N is the fast, reliable answer for all your application development. In any development language—BASIC, Pascal, COBOL, C, FORTRAN, and APL. With multikey access to records. Unlimited records per file. Duplicate, modifiable, and segmented keys.

With Btrieve/N, you can develop better network applications. And solve problems, not create new ones.

Now available for XENIX

Suggested retail prices: Btrieve, $245; Btrieve/N, $595; Xtrieve, $195; Xtrieve/N, $395; Rtrieve, $85; Rtrieve/N, $175. Requires PC-DOS or MS-DOS 1.X, 2.X, or 3.X.

Btrieve, Xtrieve, and Rtrieve are trademarks of SoftCraft Inc.;

SoftCraft Inc.

P. O. Box 9802 #917
Austin, Texas 78766
(512) 346-8380 Telex 358 200

CIRCLE NO. 201 ON READER SERVICE CARD

DECEMBER 1985

TECH JOURNAL

VOLUME 3, NUMBER 12

147

70

98

Cover Illustration · Akio Matsuyoshi

PC TECH JOURNAL

VOL. 3, NO. 12

PUBLISHER: *Newton Barrett*
EDITOR: *Will Fastie*

EDITORIAL
MANAGING EDITOR: *Marjory Spraycar*
EXECUTIVE EDITOR: *Julie Anderson*
TECHNICAL EDITORS: *Jeff Duntemann, Caroline Halliday*
ASSOCIATE TECHNICAL EDITOR: *Dan Beale*
SENIOR COPY EDITOR: *Susan Holly*
COPY EDITOR: *Gail Shaffer*
PROOFREADER: *Kathleen Peddicord*
CONSULTING EDITOR: *Thomas V. Hoffmann*
CONTRIBUTING EDITORS: *Steven Armbrust, Don Awalt, Michael Covington, Richard Foard, Ted Forgeron, Augie Hansen, Ted Mirecki, Max Stul Oppenheimer*
ADMINISTRATIVE ASSISTANT: *Paula Lamberti*
EDITORIAL ASSISTANT: *Carole Autenzio*

ART & PRODUCTION
ART DIRECTOR: *Ina Saltz*
ASSOCIATE ART DIRECTOR: *Sharon Reuter*
ASSISTANT ART DIRECTOR: *Sandra Ray*
ART SECRETARY: *Sabrina Reynolds*
ADVERTISING PRODUCTION MANAGER: *Lisa Franey Ducey*
EDITORIAL PRODUCTION COORDINATOR: *David K. Chipps*

ADVERTISING SALES
ADVERTISING DIRECTOR: *Rita Burke*
MARKETING COORDINATOR: *Julie Henderson*
ADVERTISING COORDINATOR: *Michele Fischetti*
DISTRICT MANAGERS: *Rosemarie Caruso, Jan Schultz—East Coast; Lisa Kampfmann —Midwest; Ted Bahr, Bill Bush, Phyllis Egan—West Coast*
ACCOUNT REPRESENTATIVES: *Pat Toohey, Jane Anderson—East Coast; Arlene Braithwaite—Midwest/West Coast; Pam Sigal, Jane Anderson—West Coast*

CIRCULATION
CIRCULATION MANAGER: *Shane Boel*
CIRCULATION SALES DEVELOPMENT: *Daniel Rosensweig*
MEDIA MANAGER: *Melinda Kendall*

COMPUTER PUBLICATIONS DIVISION
PRESIDENT: *Kenneth H. Koppel*
VICE PRESIDENT, Editorial: *Jonathan D. Lazarus*
VICE PRESIDENT, Production: *Baird Davis*
VICE PRESIDENT, Creative Services: *Herbert Stern*
VICE PRESIDENT, Circulation: *Alicia Marie Ivans*
VICE PRESIDENT, Circulation Services: *James Ramaley*
VICE PRESIDENT: *Hugh Tietjen*
VICE PRESIDENT, Marketing Services: *Ann Pollak Adelman*
MARKETING MANAGER: *Ronnie Sonnenberg*
BUSINESS MANAGER: *Gary A. Gustafson*
EDITORIAL DIRECTOR: *Ernest F. Baxter*

ZIFF-DAVIS PUBLISHING
PRESIDENT: *Richard P. Friese*; SENIOR VICE PRESIDENTS: *Kenneth H. Koppel, Philip Sine*; VICE PRESIDENTS: *William L. Phillips, J. Malcolm Morris, Rory Parisi*; TREASURER, *Selwyn Taubman*; SECRETARY, *Bertram A. Abrams*

EDITORIAL OFFICE
PC TECH JOURNAL, The World Trade Center, Suite 211, Baltimore, MD 21202. 301/576-0770. FAX (group 3): 301/576-9603. MCIMail: PCTECH. PCTECHline: 301/576-PCTJ. Telex: 6502565932 MCI.

ADVERTISING OFFICES
(East Coast/Midwest) One Park Ave., New York, NY 10016. 212/503-5185. (West Coast) 3460 Wilshire Blvd., Los Angeles, CA 90010. 213/387-2100; 11 Davis Drive, Belmont, CA 94002. 415/598-2290.

SUBSCRIPTION INQUIRIES
PC TECH JOURNAL, P.O. Box 2968, Boulder, CO 80321. Subscription service: 303/447-9330. Back issues: send $7.00/copy to *PC TECH JOURNAL,* Box CN, 1914, Morristown, NJ 07960.

PC TECH JOURNAL (ISSN 0738-0194) is published monthly, $29.97 for one year, $52.97 for two years, $69.97 for three years. Additional postage $12 for Canada & Foreign by Ziff-Davis Publishing Company, One Park Avenue, New York, NY 10016. Second-Class Postage paid at New York, NY and at additional mailing offices. POSTMASTER: Send address changes or subscription inquiries to P.O. Box 2968, Boulder, CO 80321.

30-Day Money-Back Guarantee

Introducing Periscope II

Professional Debugger with Break-out Switch

What Users Say:

"Very powerful for debugging and testing. Great product! Better than Atron by far."
—Wynn Bailey

[Periscope is] "the most essential element of my 'developer's tool box' . . . Every PC used for development at Microstuf has a Periscope board installed and in use on a daily basis . . . Anyone trying to write software on a PC who's not using Periscope is wasting either time, money, or both." —Jeff Garbers

Now there are TWO Periscopes! The original "board" model includes the write-protected RAM board. It is called Periscope I. The new "software" model does not include the write-protected RAM board. It is called Periscope II. We specify 'I' or 'II' only for features not in both models.

Periscope is "Always there with just a push of the button." Install the switch and software, then go about business as usual. Until your system hangs or the keyboard locks up or you just get curious about what's going on . . . then press the break-out switch and Presto! Periscope's debugging power is at your command. When you return to the executing program, it won't even know that Periscope has been there.

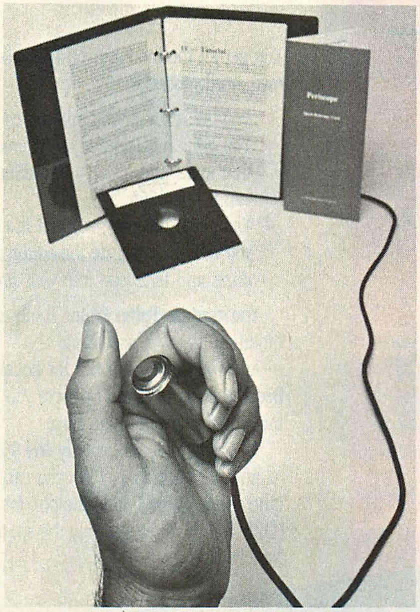
Save time with symbols and source code. They give you a roadmap through memory! Periscope uses names—symbols—from your program so you don't have to remember addresses. It displays source code and line numbers from high-level languages, too. You save hours of time because you access what you need with familiar names!

Can debugging be fun? With Periscope it can! Here's why:

It's Fast! Written entirely in assembler to save you time.

It's Easy! Commands similar to Debug's, optional on-line help, quick-reference card, tutorial, 150+ page manual, and support direct from the author—all you need to get up to speed quickly.

It's Flexible! You can:
- Enter Periscope via break-out switch, program loader, or assigned keys
- Define up to four windows for data, register, stack, and disassembly information; change them any time
- Display memory in ASCII, byte, word, double word, integer, or signed integer
- Design your own templates to display memory in easy-to-read formats
- Customize Periscope with your own programs via the user exit capability
- Assign frequently-used commands to function keys



The break-out switch really sets Periscope II apart from typical software-only debuggers. It installs easily, without taking an extra slot.

It's all there! Periscope includes all the standard debugger features, plus extras like supporting the use of one or two monitors, enabling you to search for address references, etc. The latest version includes new and enhanced features to help you debug your programs faster and easier than ever before:

New!	In-line Symbolic Assembler
Enhanced!	75+ Breakpoint Options
New!	Traceback
Enhanced!	Optional Windows
New!	Source-Level Debug
New!	View text files
New!	EGA Support
New!	User Exits
New!	8087/80287 Status
	Much More!

Periscope's unique breakpoint power forces bugs out from where they hide! With over 75 breakpoint options, including both temporary and sticky code breakpoints, you'll find elusive bugs fast. For instance, you can break on register, byte and word values; stop on execution of source lines and interrupts; break on reads and writes to ranges of I/O ports and ranges of memory using various tests. The memory breakpoint is great for debugging C programs with broken pointers. You can even write your own custom breakpoint tests!

Periscope requires: An IBM PC, XT, AT or close compatible; DOS 2.0 or later; 128K RAM; Disk Drive; an 80-column Monitor.

What Reviewers Say:

"This product is a reliable and useful tool for any programmer's workshop. A number of innovative, helpful features are yours for the taking." —Programmer's Journal

"Periscope has excellent on-line help. We were impressed by Periscope's very fast response in all its operations. It is a pleasure to use, and a refreshingly different product . . . offers great value and unique advantages." —Boston Computer Society

Debug any program, any time! Periscope is, in one user's words, "Robust". Use it to debug almost any program, even device drivers, memory-resident, and non-DOS programs. Debug when DOS is broken; debug DOS. Periscope won't let you down when you need it.

User calls Periscope I "Bulletproof"! The installation program in Periscope I loads the crucial debugger software into the RAM board and write-protects it. No runaway program can touch this code! Coupled with the break-out capability and the built-in reliability of the software, this protection gives you the most powerful crash recovery system available.

"Outstanding value for the services offered" writes one user. ***"Worth the price!"*** writes another. With the 30-Day Money-Back Guarantee, you can't lose! If you're not satisfied with Periscope for any reason, just return it within 30 days of receipt for a full refund.

Periscope I: Board, Switch, Software, Manual, Reference Card . . Just $295.00

Periscope II: Switch, Software, Manual, Reference Card . . . . Only $145.00

The U.S. Navy gets Periscopes from us . . . shouldn't you? Order today!

For Immediate Shipment Or More Information, Call Toll-Free

800-722-7006

Get Your Programs Up and Running;

UP PERISCOPE!

DATA BASE DECISIONS
14 Bonnie Lane • Atlanta, GA 30328
404/256-3860

CIRCLE NO. 215 ON READER SERVICE CARD

60-DAY MONEY-BACK GUARANTEE

Introducing Borland's New Turbo GameWorks™ $69.95

THE GAMES YOU CAN PLAY, REPLAY, REVISE AND REWRITE BUT CANNOT RESIST. (Turbo Pascal Source Code included!)

We give you the source code, the manual, the diskettes, the 60-day guarantee and the competitive edge. Let the games begin. Chess. Bridge. Go-Moku.

State-of-the art games that let you be player, referee, and rules committee—because you have the Turbo Pascal source code. Which means that you can play a game or create a game, any time and any way you want.

Borland's new Turbo GameWorks lets you combine gamesmanship with craftsmanship. Discover the secret techniques and moves used by the Old Masters. Learn exactly how state-of-the-art computer games are made—so you can go off and make your own. Since you have the source code, you can always change the game. Or rig the game, if no one's looking.

Pure Magic. That's Turbo GameWorks. And part of the "sourcery"—Turbo GameWorks is only $69.95. When combined with our new Turbo Editor Toolbox in the Turbo *New* Pack, it's only $47.50.

SHORT CUTS, SECRETS AND STRATEGIES. The Turbo GameWorks manual takes you step-by-step through all the games. How to play them. How to modify them. How to use the power of Turbo Pascal to write new games.

You'll learn general problem analysis, how to identify all possible moves, "rule of thumb" strategies, procedures for testing strategies, and ways to rate options. You'll also be introduced to "top down" program design, the development of basic algorithms, the use of constants and data structures and ways to design short cuts with incremental updating.

On top of all that, you'll have a lot of fun (if you want to).

So go to play (and work) with Borland's new Turbo GameWorks. It's unique. It's fascinating. And it's brand-new. Be first with the latest and greatest—order your Turbo GameWorks today.

BORLAND'S TURBO GAMEWORKS AND SOME OF ITS MASTER PIECES

Chess, the ultimate strategic game. A game so old that no one knows its exact origins. Turbo GameWorks lets you play chess at six different levels from the beginner to the sophisticated user. And you have many ways of playing with your Turbo GameWorks. Let the computer solve checkmate problems. Set the time limit for each game. And there's more.

Decide whether you or the computer "goes first." Trade places with the computer at any point in the game. It's all possible with GameWorks.

Go-Moku, also known as "Five-in-Line," is a very old Japanese game played on a board of squares. The first player to get five game pieces in a row—either horizontally, vertically, or diagonally—wins the game.

It's an intriguing game. But you're not limited to playing it one way. With Turbo GameWorks, you can modify it your way.

Bridge. Play bridge with a friend or team up against the program—you decide which hands the computer plays. You can even decide to let the program cheat! The program automatically bids and plays its own hands. And, since you can tinker with the source code, you can make "your" Bridge unlike any other.

4585 Scotts Valley Drive, Scotts Valley CA 95066
Phone (408) 438-8400 Telex 172373

TURBO HOLIDAY JUMBO PACK $245.00

Now you get the whole Turbo Pascal family, including its newest members, at an incredible price.

- Turbo Pascal 3.0 combines an integrated programming environment with the fastest Pascal compiler in the galaxy.
- Turbo Tutor teaches you step-by-step how to use the Turbo Pascal development environment. With commented source code for all program examples on diskette.
- Turbo DataBase Toolbox gives you all the tools you need to sort and search your data and build powerful database applications.
- Turbo Graphix Toolbox™ gives you a complete library of Pascal procedures to build applications using high resolution graphics and graphics window management.
- NEW! Turbo GameWorks—Chess, Bridge, Go-Moku. Secrets and strategies of state-of-the-art computer games are revealed with complete source code.
- NEW! Turbo Editor Toolbox—the Pascal modules that let you build your own word processor. A full-featured WordStar-like program, MicroStar™, is included on your disk.

You're getting everything at only $40 each. And if you already own one or several members of the Turbo family, be creative—nothing can stop you from buying the Jumbo Pack, picking out the ones you already have and giving them as holiday gifts for family or friends. At these prices you can afford to give to others and to yourself.

With Dazzling Discounts!

CIRCLE NO. 250 ON READER SERVICE CARD

NOT COPY-PROTECTED

Holiday Gift Packs, Turbo GameWorks™ & Turbo Editor Toolbox™

Available at better dealers nationwide. Call (800) 556-2283 for the dealer nearest you. To order by Credit Card call (800) 255-8008, CA (800) 742-1133.

		Quantity
*Turbo Holiday Jumbo Pack	$245.00	____
*Turbo Holiday Pack	$125.00	____
*Turbo New Pack	$95.00	____
Pascal	$69.95	____
Pascal w/8087	$109.90	____
Pascal w/BCD	$109.90	____
Pascal w/8087 and BCD	$124.95	____
Turbo DataBase	$54.95	____
Turbo Graphix	$54.95	____
Turbo Tutor	$34.95	____
Turbo Editor	$69.95	____
Turbo GameWorks	$69.95	____

These prices include shipping to all U.S. cities. All foreign orders add $10 per product ordered.

*Gift Pack Offers Last Until March 31, 1986

The holiday packs include an upgrade coupon for both options so you get BCD and 8087 support for $39.95 (regularly $55.00).

Carefully describe your computer system!

Mine is: — 8-bit — 16-bit

I use: — PC-DOS — MS-DOS — CP/M-80 — CP/M-86

My computer's name and model is: ____________

The disk size I use is: ☐ 3½" ☐ 5¼" ☐ 8"

Name: ____________

Shipping Address: ____________

City: ____________

State: ________ Zip: ________

Telephone: ____________

Amount: (CA add 6% tax) ____________

Payment: VISA MC Check Bank Draft

Credit Card Expiration Date: ____ / ____

Card # ____________

NOTE: Turbo Editor Toolbox and Turbo GameWorks are available for the IBM PC and true-compatibles using Turbo Pascal 3.0 ONLY.

COD's and Purchase Orders will not be accepted by Borland International. California residents: add 6% sales tax. Outside USA: add $10 and make payment by bank draft, payable in U.S. dollars drawn on a U.S. bank. H6

Livin' in the D.O.S.

Users deserve more than developers are giving them.

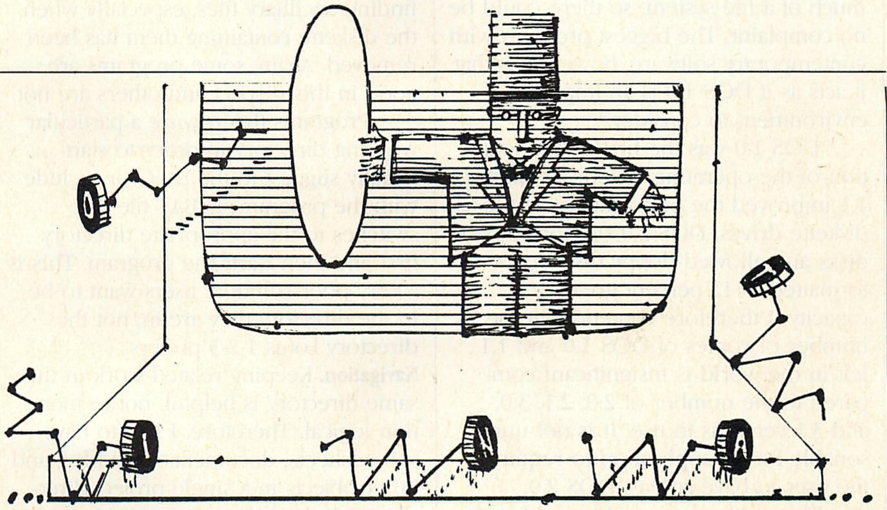


ILLUSTRATION • MACIEK ALBRECHT

We are well past the fourth birthday of the IBM PC. DOS 2.0 soon will be three years old. By computer industry standards, that should have been enough time for software developers to have learned a few good habits. By my reckoning, most have not, and this makes me mad: first because I am an end user who wants and needs quality software, and second because I am a software developer by training and know what can be done.

This is, therefore, a laundry list of my complaints about contemporary software products for the IBM family of desktop computers. It comes from years of coaxing developers to consider these issues, from years of having only a few listen, and from years of using programs that are just a lot of trouble.

VIDEO MODES

The first place that most programs seem to falter is with the display system. I know of no program that decides for itself how to handle the display.

Determining which displays are present is possible to do with software. IBM does it with its diagnostic programs. Nonetheless, most programs require the user to define the display characteristics. Some even fail to set the display to a text mode, if that is what they require, and therefore often can produce unreadable garbage if they are invoked from a graphics mode. Such programs assume that the machine will be in the desired mode, or that the user will set it manually.

Programs should assume nothing, and they should make the display analysis on their own. The user need be consulted only to determine which of multiple displays should be used as the default and, in the case of the Color Graphics Adapter, whether snow is acceptable. The list of displays to look for is blessedly short: the CGA, Enhanced Graphics Adapter, IBM monochrome, and the Hercules card probably account for 90 percent of all installed systems. Most other boards emulate at least one of those standards (usually CGA or Hercules) and therefore are not a problem.

Recently, I was testing a brand new, $700 program. When I reached the point at which the video mode could be selected, I chose the monochrome option to see how it would react with my EGA. The program had to be reinstalled before it could write to the screen again. Was I impressed?

THE KEYBOARD

Catastrophe approaches. More and more programs are being written as resident programs that intercept the keyboard, process their own keystrokes, and pass the rest on. Many of the programs are quite useful, but using more than one at a time is asking for trouble. In fact, using only one can sometimes create major interference with a nonresident application such as a word processor; the application also may interfere with the resident program.

To be fair, the real job of handling the keyboard falls to the hardware and the operating system. DOS does, in fact, provide the facility to change an interrupt and, therefore, to direct the handling of external events to a different program. Designers, however, did not expect programs to stack keyboard handlers on top of each other; no facility exists to manage multiple handlers.

Worse, most programs do not expect to have their keystrokes preprocessed, nor do they expect an underlying program to need keystrokes that are not meaningful to the running program. Most applications do not pass keystrokes along to lower programs, effectively creating the rule that the last loaded program owns the keyboard.

The problem seems to have no general solution. The PC software industry would do well to adopt some conventions. Multiple programs living together should communicate with one another to assure that all keystrokes are properly handled—and by the proper program. For example, resident programs should accept a key sequence that turns off their trapping and allows all but one sequence to be passed through to the next layer. Perhaps Alt-Shift could be the "turn-trapping-off" signal and Alt-Ctrl could be the "resume-trapping" sequence. Multiple Alt-Shifts could thus turn off trapping, moving down the stack of programs, while Alt-Ctrl could climb back up. These sequences would have to be avoided by applications, but other keystrokes that might have been trapped by, say, a keyboard macro program could be passed instead to the actual application.

In the future, perhaps the operating system will provide the capability of stacking keyboard vectors so that an arbitrary number of programs can be stacked. Even then, interprogram communications standards will be needed.

FILE SYSTEM

Most software for the PC still exists in the dark ages when it comes to proper handling of the file system of DOS 2.0 and later. Earlier versions did not have much of a file system, so there could be no complaint. The biggest problem with contemporary software, however, is that it acts as if DOS 1.0 is an important environment to consider.

DOS 1.0 was the first implementation of the operating system. Version 1.1 improved the performance of the diskette drives. DOS 2.0 supported hard disks and allowed floppy disks to be formatted to 12-percent greater capacity. I therefore contend that the number of copies of DOS 1.0 and 1.1 left in the world is insignificant compared to the number of 2.0, 2.1, 3.0, and 3.1 versions in use. It is not unreasonable for an application to require the user to have at least DOS 2.0.

That said, what are the problems? First, starting a program and, second, navigation in the file system.

Start-up. DOS provides two methods of starting a program in DOS 2.x and a third in DOS 3.x. Most simply, the name of the program is typed on the command line and, if the executable file is in the working directory, the program will begin to run. DOS also provides the search list (the result of the PATH command); this is simply a list of directories in which DOS should search for the program if it cannot be found in the working directory. The intent of PATH is to allow programs to be executed from other directories. In DOS 3.x, the program also may be executed by typing a complete path name instead of just the program name:

```
C:\DOS\PROGRAM
```

These good features are sufficient to allow many programs to be started. But many other programs, especially the big, expensive applications, need to have additional files in order to operate correctly. These files usually are stored with the executable program, so when the program is run from another directory it will not see its support files in the working directory. In versions 1.0 and 2.0, DOS provided the programmer with no help in finding the name of the directory from which the program had been loaded. This was solved in 3.0. However, a few programs are, in fact, able to find themselves in DOS 2.0. Applications that work this way are a real joy and require minimal installation. Even so, the reactions have been mixed when I have suggested to a number of firms strategies for doing this.

While the above discussion focused on hard disks, the same strategies can also be applied to floppy-disk systems. In those cases, the biggest problem is finding ancilliary files, especially when the diskette containing them has been removed. Again, some programs are good in this respect and others are not.

Programs that *require* a particular working directory in order to start usually suggest to the user (or include with the program) a .BAT file that switches to the appropriate directory first and then starts the program. This is a very poor solution; users want to be in the directory they are in, not the directory Lotus 1-2-3 prefers.

Navigation. Keeping related work in the same directory is helpful, not to mention logical. Therefore, I like to have spreadsheets, documents, data files, and other objects in a single project directory. Generic files can go anywhere, but storing them with the program that uses them is quite clumsy. On the other hand, files that the program itself may generate and that are not specific to my work should be stored with the program and out of my way.

That is the primary reason for starting programs from a particular directory. Nonetheless, I often must move around the directory structure from within the program to find other files. A common case is that of a standard memo or spreadsheet format to be used for a new document or analysis.

Most programs I have used are painfully awkward at navigation. The designer seems to assume that DOS provides file system navigation at the command level, so inclusion in the program is redundant. Dropping out of an application, particularly one that requires start-up in a specific directory, to perform navigation is very time-consuming. Even an environment such as Microsoft Windows is no help: although the user can switch context to a file management system, directory selections do not alter the settings for the application. Applications need to provide a great deal of assistance to the user.

The vast majority of programs I have tried have no facility even for changing directories. A minority allow path names to be specified when file names are requested. Some, such as Lotus 1-2-3 and Ansa's Paradox, require two steps. First, the directory must be changed, and then a simple file name (one with no path information) can be specified. Although 1-2-3 and Paradox are helpful when displaying lists of files from which to select, neither one provides this courtesy for directories.

In short, navigational aids are limited or nonexistent in most programs. This is simply a result of laziness: with minimal amounts of coding, navigational aids can be implemented—at great benefit to the user. Always a lame excuse, the amount of RAM available is even less of an excuse now.

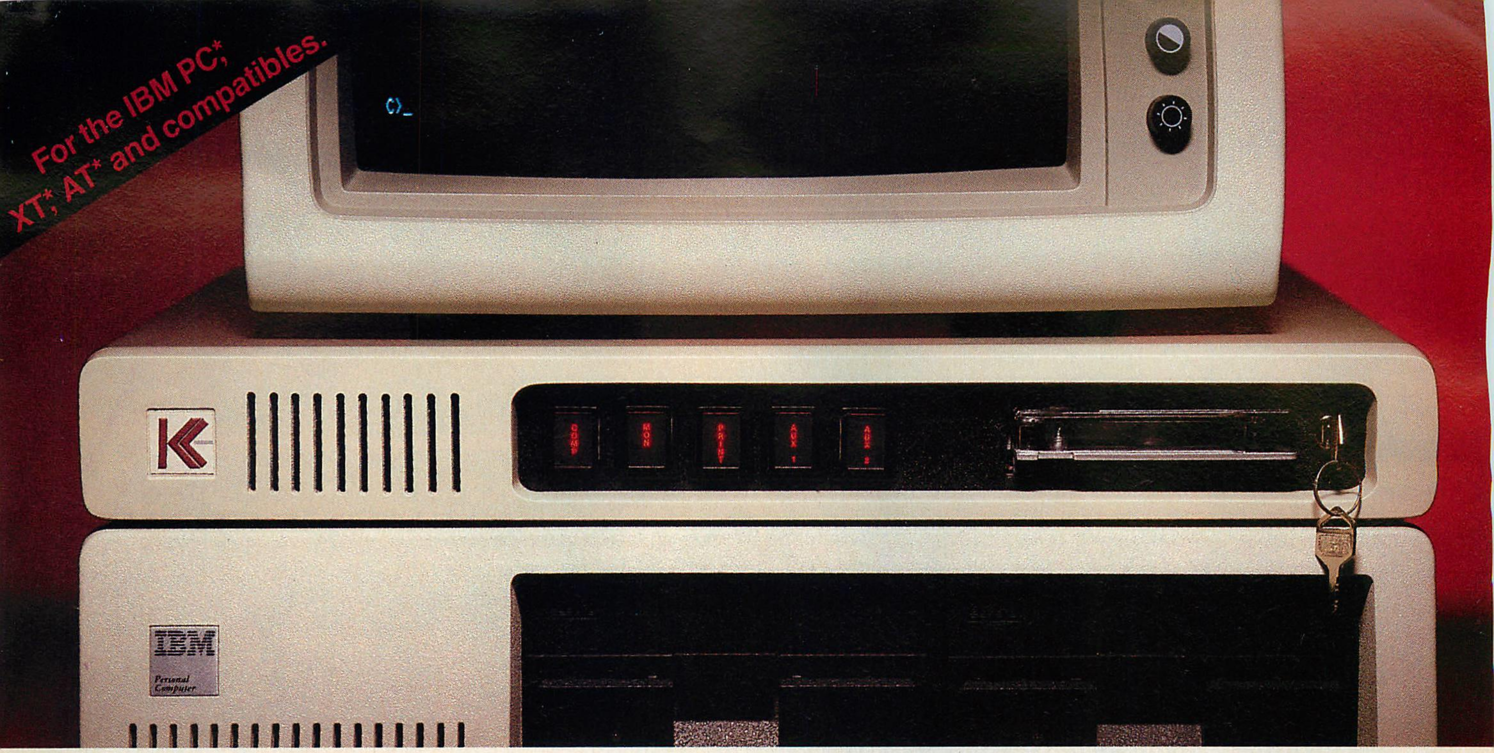
The products from Satellite Software, WordPerfect in particular, are shining counterexamples. WP finds itself, its ancilliary files, and has full navigational capabilities. The user may never have to exit WP except to format new floppies. It is a worthy model.

Printers. Program designers hardly can be faulted for experiencing total frustration in dealing with printers. About all that the programmer can know for sure is that the printer will be connected via a serial or parallel port. Today, it is also fair to assume that most printers can perform a backspace and do not automatically produce a line feed when they see a carriage return. Unfortunately, the operative word is *most*; programs still have to be prepared for printers that cannot perform those simple functions.

Frustration or not, applications should nonetheless behave responsibly toward printers. Two examples spring to mind. The first, again, is SSI's WordPerfect, which in its current version is delivered with 191 printer definitions *and* a program that lets the user further customize if SSI's choices are not ideal for his/her circumstances or if, by some remote chance, SSI has omitted the user's printer. The other end of the spectrum is represented by two programs from IBM's Personally Developed Software series: Checkbook I/O and PC Print. These programs are nicely done, but *require* an IBM printer (they use the extended character set). An Epson FX-80 emulation is not sufficient.

Adding features of this kind to applications is certainly not without cost. Designing 191 printer definitions cannot be done overnight. What is frustrating is that so few programs make the attempt; support for path names, for example, is not a major project.

The customer, the ultimate end user, will appreciate this level of sophistication. Let's get cracking.



We put hard disk storage and backup where it belongs.

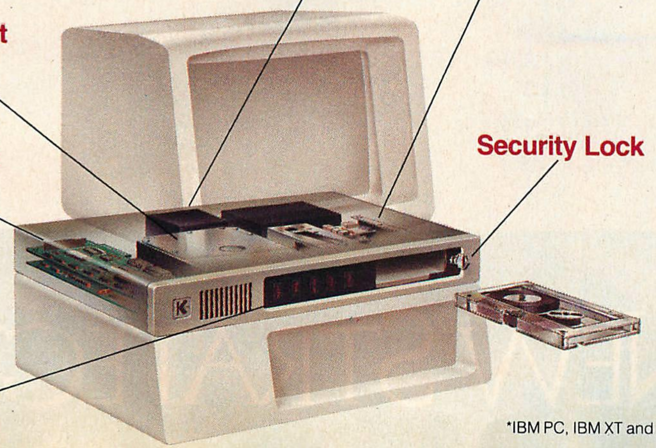
Maximum storage in a minimum space.
A mass storage solution for personal computers need not occupy massive amounts of space.

That's why Kamerman Labs packs an entire hard disk storage and backup system into a package only 2½ inches high.

In fact, our Masterflight system is so well integrated with your PC, it looks like it should have been there all along.

Hard disk storage.
Inside the Masterflight is a high-performance hard disk system, available in 10, 20 or 30 Megabyte versions. You get up to 100 times the storage capacity of floppies. At up to 10 times the speed. Spend more time working and less time waiting.

Streamer tape backup.
Any professional working environment demands a reliable backup solution. The Masterflight provides it through a high-performance streamer tape that backs up an entire 10 Megabytes in less than 4 minutes. Streamer tapes are available in 20, 40 and 60 Megabyte capacities. Instead of just a hard disk, Masterflight gives you a complete data storage solution.



Power control.
Masterflight directs power to your computer, monitor, printer and two other devices through front panel switches. No more blind groping on back panels to power up your system.

Surge protection.
A little power surge can produce a lot of damage. Masterflight protects your system with special circuitry that filters out spikes, surges and line noise.

Security.
Masterflight guards access to your system through a fully lockable power switch.

Phone today. Dealer and OEM queries welcome. 800-522-2237
In Oregon, Hawaii and Alaska call 503-626-6877

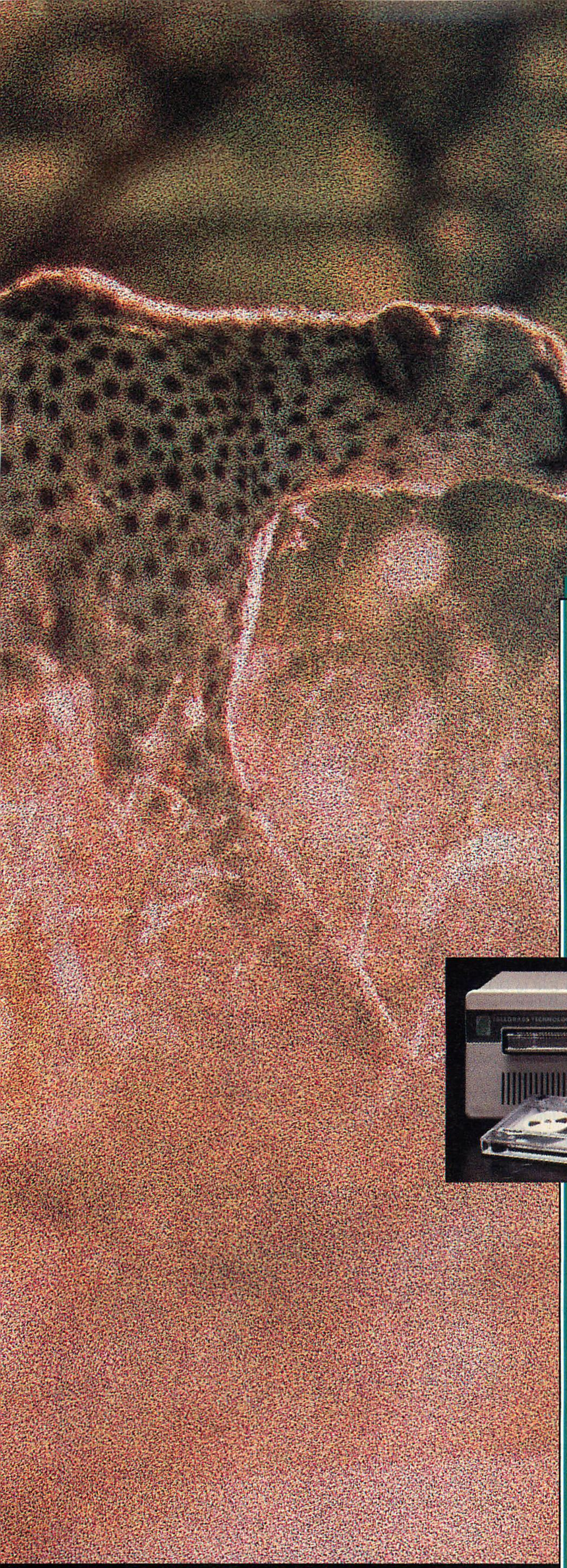
Kamerman Labs

8054 SW Nimbus, Bldg. 6
Beaverton, OR 97005
Phone 503-626-6877

CIRCLE NO. 244 ON READER SERVICE CARD

*IBM PC, IBM XT and IBM AT are registered trademarks of International Business Machines.

A SLEEK NEW STRATEGY FOR



Tallgrass ends the hunt for the perfect mass storage system.

Until now, all Personal Computer tape backup systems have seemed unnecessarily cumbersome and complex. Then along came PC/T,® from Tallgrass.

PC/T stores and retrieves data on tape in a way that makes all streaming tape systems obsolete. Because it makes tape work like disk. With DOS simplicity. Rapid access to the file you need. And uncompromised data integrity.

The key: PC/T *formats* tape, creating directories, just like with floppies or hard disks. So defective blocks are locked out. And all 60 megabytes of tape are on line, to be accessed, updated or deleted at the stroke of a key. And if recorded data is accidentally damaged, PC/T's error correction capabilities can completely reconstruct it.

PC/T is available in a wide range of Tallgrass systems, offering tape only or tape plus 25 to 80 megabytes of hard disk storage. Just call 1-800-228-DISK for your Tallgrass dealer. And end the hunt for the perfect mass storage system.

For a framable print of the Acinonyx jubatus *Cheetah, send a check for $3 to: The Tallgrass Collection, Dept. C, 11100 W. 82nd St., Overland Park, KS 66214. Proceeds, plus an additional donation from Tallgrass, will be forwarded to the World Wildlife Fund.*

See Us At COMDEX

TALLGRASS® TECHNOLOGIES

WHEN IT'S WORTH SAVING.

CIRCLE NO. 247 ON READER SERVICE CARD

TAPE: PC/T

BRIEF™

"BRIEF, The Programmer's Editor, is simply the best text editor you can buy."

John Dvorak, InfoWorld 7/8/85

Choosing the right program editor is an extremely important decision. After all, more time is spent with your program editor than with any other category of software. A good program editor helps to stimulate creativity and produce programs of superior design. A poor program editor, on the other hand, fights you every step of the way. It can slow you down, get in your way, and make even the simplest tasks difficult.

BRIEF - PROGRAM EDITING *YOUR WAY*

Every programmer has an individual style that makes their work unique. Most program editors compromise that style by forcing the programmer to conform to their methods. Not so with BRIEF.

BRIEF's most powerful feature is its ability to conform to your way of programming. BRIEF can be easily tailored/customized to your individual preferences. For example, keyboard reassignment allows the keyboard to be used in whatever way you prefer. Keystroke macros allow long sequences of editor commands to be repeated by pressing a single key. Plus you can simultaneously work with numerous program and data files, and configure many windows to control BRIEF's visual presentation.

EXPERT USERS OF EVERY MAJOR EDITOR HAVE SWITCHED!

(Call For Details)

"A BONA FIDE UNDO" Steve McMahon – BYTE REVIEW "BUILD YOUR DREAM EDITOR" MARCH 1985

Here are Steve McMahon's exact words: "...BRIEF implements a true undo facility, by default allowing command-by-command recovery from the last 30 commands...The number of commands you want to be able to undo can be changed." (up to 300) "Only with BRIEF, though, was it possible to undo a macro that produced 4000 words of text with a single keystroke."

EVERY FEATURE YOU WANT

BRIEF supports practically every feature you've ever seen, and some you probably haven't thought of yet.

- Edit Multiple Large Files
- Windows (Tiled & Pop-up)
- Unlimited File Size
- Full UNDO (up to 300)
- True Automatic Indent for C
- Exit to DOS Inside BRIEF
- Compile programs inside BRIEF
- Uses All Available Memory
- Online Help
- Tutorial
- Repeat Keystroke Sequences
- 15 Minute Learning
- Reconfigurable Keyboard
- Adjustable line length (up to 512)
- Search for "Regular Expressions"
- Mnemonic Key Assignments
- Horizontal Scrolling
- Comprehensive Error Recovery
- And ... a Complete, Compiled, Programmable and Readable Macro Language.

```
================================ make.c ================================
int       handle = 0;

main (argc, argv

int            argc;
============== fsa.h ============================== makefile.h ========
#include "..\include\ctype.h"          |/**
                                       | **          makefile.h:
typedef  struct                        | **
{                                      | **       This is the definitions fil
   short    action,                    | **   Hopefully, it won't be unreasonab
            state;                     | **   that have been written.
} Fsa;                                 | **/
#ifdef   FSA_MAIN                      |
Fsa      fsa[][8] = {/* Alphanum    Co |typedef struct cmd_struct
                                       |{
/* State 0. */         0, 2,        10 |   char        *cmd_text;
/* State 1. */         10, 0,       10 |
/* State 2. */         0, 2,        1, |   struct cmd_struct *next_cmd;
/* State 3. */         0, 5,        11 |} *Cmd_Ptr, Cmd;
/* State 4. */         0, 4,        0, |
Mismatched open parenthesis.               Line: 11   Col: 17     2:17 pm
```

A TYPICAL BRIEF SCREEN

Notice there are three windows on the screen simultaneously and each one is viewing a different file. The mainline of a C program is visible in the uppermost window; the programmer has run a syntax checking macro which found a mismatched open parenthesis in the arguments to the mainline. The other two windows show header files containing information crucial to the design of the program. BRIEF can have an unlimited number of windows and files accessed simultaneously.

BRIEF'S ONLY LIMITATIONS ARE THE ONES *YOU* SET

AVAILABILITY: IBM-PC, AT, Compatibles & TANDY 2000

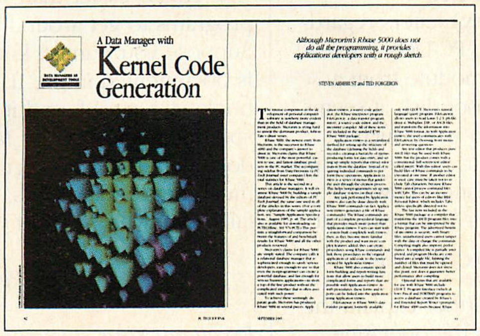
Try BRIEF with a full 30 day moneyback guarantee. Only $195. Call 800-821-2492.

Solution Systems™

335-P Washington St., Norwell, MA 02061 (617) 659-1571 **800-821-2492.**

Steve McMahon's quote courtesy Suntype Publishing Systems. BYTE review by Mr. McMahon may be found in BYTE Magazine March 1985.

CIRCLE NO. 123 ON READER SERVICE CARD



CUTTING THROUGH LAYERS

Art Krumrey's article ("SNA Strategies," July 1985, p. 40) grossly distorts SNA in attempting to fit it into the inapplicable ISO OSI RM (International Standards Organization Open Systems Interconnection Reference Model).

First, SNA is divided into only five layers; both end user and physical are out of scope. Of these five, DLC (data link control) is a fence straddler, only marginally considered part of SNA, while the remaining four do not cleanly correspond to OSI RM layers. Further, SNA has components such as PU, PUCP, and SSCP which cut across layers. Finally, IBM has assigned only names, not numbers, to SNA layers.

As a side note, the non-SNA text is also inaccurate—V.21 should have been X.21 and V.35 was omitted.

In discussing PC options, the author misses several obvious classes of products. SNA RJP (remote job processing) products for SLU (single logical units) and MLU (multiple logical units) are available from both IBM and third-party vendors, usually concurrently with LU2 (3270) sessions. The 3270-PC with DFT support offers multiple sessions, and third-party software supports DFT on the standard PC, PC/XT, and PC/AT.

Seymour J. Metz
Annandale, VA

Differences exist between the two elements of any analogy. That is why it is an analogy and not an equivalence. Perhaps the words "compare with" would have been more precise than "correspond." In fact, the word "corresponding" and the analogy itself come from "SNA and Emerging International Standards." IBM Systems Journal, *volume 18, number 2, 1979.*

I appreciate your correction that V.21 should be X.21.

The 3270-PC is a "coaxial connection to the IBM 3274 or 3276 controller" and thus is a special—and more expensive—case of my method 1. The fact that the 3270-PC allows multiple logical units—multiple simultaneous sessions—does merit mentioning. It also has been suggested that the 3270-PC may use LU 6.2 in the future.

—Art Krumrey

While I was glad to see SNA acknowledged in the PC literature, I feel it necessary to comment on Art Krumrey's article. Some statements might mislead readers unfamilar with IBM mainframe communication techniques.

First, SNA is not a "cable-and-controller infrastructure," as was stated in the article, any more than any other data communications scheme. I think it is safe to say that SNA specifications relate more to message structure and exchange than to hardware support (at least from the end user's perspective). In fact, a typical PC-to-IBM mainframe connection requires the same overall (controller and cable) hardware support for the familar ASCII/asynch protocol as for SNA/SDLC. The appropriate option cards and modems do, of course, differ but the major difference is really in the software, both on the mainframe and on the PC. The cabling involved is identical on both ends.

However, the major clarification I wish to make is with respect to the second method listed for connection of a PC to an SNA mainframe. It is implied that a dedicated 37X5 port and a leased telephone line are required. Such is not the case. Ordinary Bell-compatible synchronous modems on the mainframe end may be connected to single voice grade lines or rotaries in order to receive incoming calls. Synchronous modems for the PC end may be purchased for around $700 at 2400 bps.

Even given the above, SNA/SDLC connection is more expensive than its alternatives, but, considering the overall costs of professional data processing and fully configured PCs, I believe that the operational advantages of this mode of connection justify its expense. In view of the rapidly evolving field of intersystem connectivity (at least partially in preparation for which SNA has been developed), its place seems unquestionably established for PCs in an IBM mainframe environment.

Campbell S. Nelson
Los Angeles, CA

I agree that "SNA is not a cable-and-controller infrastructure any more than any other data communications scheme." What is significant about SNA, however, is that it is a ubiquitous infrastructure in corporate America—that is of great value to corporate PC users.

The second method of PC connection does require a dedicated 370X port in almost all cases. Depending upon the size of the existing 370X, this $3,000 to $7,000 cost is substantial. Dial lines work, but have much higher error rates than digital data service lines.

Whether switched or dedicated lines are used, the importance of SNA is its future capabilities (LU 6.2) and that it allows full use of existing facilities.

—Art Krumrey

R:BASE SECURITY

Thanks for the detailed review of R:base 5000 ("A Data Manager with Kernel Code Generation," Steven Armbrust and Ted Forgeron, September 1985, p. 82). I have been using R:base since it was a pup (versions 4000 and 5000) and your reviewers seem to know at least as much about it as I do. I would like to point out that Microrim has just released an upgrade that allows 64K instead of 4K characters in a WHILE loop. Their timing was good—I was running into the message "Too many commands in WHILE loop" almost daily. They also added a decimal alignment formatting mask for use with reports.

R:base does have one important "misfeature" that I hadn't known about

until recently. The PASSWORD protection is *not* encryption. The data are still present on the disk, protected only by Microrim's "unique file structure." This file structure can be broken. I accidentally REMOVEd my most important table of information. If I had been using, say, Paradox, I could have simply un-erased the appropriate file. With R:base, the data were present on disk yet unavailable. I called Microrim for help—surely I could get back the data by simply resetting some pointers in the file!? The technical helpers were most apologetic about the fact that their file structure is proprietary and secret.

It took me eight hours to get the (password-protected) data back this first time using a specially created Turbo Pascal program; with practice, however, that time undoubtedly could be cut down. I feel that I can no longer depend on R:base for security.

In the same vein, CLOUT version 1.0 totally ignored R:base password protection. The first thing I did was to type the **eed** in version 2.0.

I heartily approve of your longer, more technical articles. If the September issue reflects a trend, I say bravo.

Neil J. Rubenking
San Francisco, CA

UNNECESSARY ZAPPING

Having read the (many) letters regarding the ability to have PC-DOS 2.x read its COMMAND.COM module from a disk other than the boot disk, it seems somewhat pointless to write such a complex routine as Mr. Nisley has done ("A Change in Command," Edward Nisley, July 1985, p. 149). Instead, put

```
SHELL = C:COMMAND.COM C: /P
```

in CONFIG.SYS. To further explain the shell option, SHELL is the operating COMMAND.COM though you need not call COMMAND.COM by that name if you do not wish to. The command parses out to be (after the equal sign):

C:COMMAND.COM—Where to find the command processor and what it is called.

C:—When reloading the command processor, get it from here. If no name is given, then the system uses the same name given for the initial load (two different COMMAND.COMs, an initial one and, in addition, one that does the work, for example, after a LOGIN to the system for security).

/P—Tells COMMAND.COM that this is the initial loading of COMMAND.COM and that it must perform certain housekeeping steps.

This is easier than writing complex programs to get around the problem. Suppose you want to know where this is documented in the IBM DOS 2.x books. It isn't; I found it in the Microsoft documentation for MS-DOS 2.x, a document that for programmers is much better than IBM's, but for users falls somewhat short of the mark.

Peter D. Hipson
West Peterborough, NH

Excellent! Your suggestion is better than the other method I've seen—typing

```
COMMAND C:\ /P
```

at the DOS prompt. The COMMAND statement relocates the command processor and it works with a RAM disk (SHELL does not). But it invokes a second command processor and costs you 3,088 bytes of RAM. If you use COMMAND in an AUTOEXEC.BAT file, watch out for infinite loops. DOS will restart the AUTOEXEC sequence when it encounters COMMAND. Change the default directory before the COMMAND statement to avoid this.

It turns out that the SHELL command is discussed on page 9-10 of the DOS 2.0 manual. The /P option is not mentioned and, should you leave it off,

AUTOEXEC.BAT isn't executed and the EXIT command runs off.

The first version of COMZAP just found the right byte and changed it. Even that was more complex than using SHELL, had I only known . . .

—Ed Nisley

NONSUPPORT SUPPORT

I seldom write about what I read in magazines, but I am writing to echo the comments made in "Directions" by Will Fastie ("To Support or Not to Support," August 1985, p. 9).

He was being rather gentle, I think, with IBM, according to my experience. I must tell you that, to put it mildly, I am not at all happy with IBM support because I don't think it exists. It's no use calling your dealer because every dealer that I have called is abysmally ignorant when it comes to the IBM Personal Computer particularly. Calling IBM generates vague replies or promises to call back—and no response. While I was unhappy with IBM in the beginning and bought an IBM computer only under duress, I must tell you that I am even more unhappy now, although not with the machine. I am perhaps more satisfied with it than I am with the so-called service that IBM provides.

I must also echo and commend Apple support, which in the beginning was tremendous, then flattened out and came to a very low ebb, but has gradually come back to a more substantial level. I am quite satisfied. The response time from Apple is very good and the firm is interested in helping.

Perhaps the problem with IBM is that the company is too large. IBM has been getting away with anything because of its position in the market and the amount of money and clout it wields. However, by not providing support to the broad PC base that is out there, IBM has begun a small hemorrhage in its company that could trickle its life away. If this is the firm's philosophy, then I hope it does trickle the life away because IBM does not deserve to stay in the marketplace if it is not going to be tuned to the needs of all its customers, not just the big spenders.

Leo D. Bores, M.D.
Scottsdale, AZ

I appreciated your last "Directions" column because we are small developers of software for "MS-DOS machines" as we prefer to call them. We don't like to name company names to indicate we prefer one brand or the other.

We try to minimize hardware dependencies (just like everyone else) in order to maximize portability. The one area, however, where we must be hardware dependent is the serial ports due to the requirements of driving special serial devices in an extremely interactive situation. All we need are the addresses of the various ports (input, output, line status, line control, and so on) and the bit assignments. It provides a very good test of various companies' abilities to find straightforward information. A little obscure, perhaps, but very concrete information.

To this day, we have photocopied pages of the serial port details from the IBM second technical reference. We have the first manual, which I like very much. But we can't get the second one.

Compaq or Computerland personnel were no better because they just couldn't find the information. But they did everything they could to get our software onto the Compaq series. And our software ran perfectly, and fast.

As a result of a customer's request, I had to investigate the port details for an AT&T 6300. I found the telephone number in the phone book; they gave me a technical hot-line 800 number and I called. It took three minutes at the

most. Of course, the technical people were busy, but they called me back 20 minutes later. Highly unusual. I did not expect that, although I was curious. It took the person five minutes of *his* phone time to find the information in his manual. (Yes, the addresses and so on are the same). I was impressed.

Why is information for software developers buried so deep? Why is it so hard to get to technical people for answers and where are the technical manuals? How long will AT&T provide such good service? Until they dominate the market like IBM?

Patrick B. Cawood
Western Hydrologic Systems
Los Angeles, CA

A HELPFUL DELAY

Your July 1985 issue an article called "Power Graphics" (Thomas V. Hoffmann, p. 56), which compared the IBM Professional Graphics Controller with the Vectrix VX/PC, was brought to my attention by the engineering department here at VMI. With respect to the IBM PGC device driver listing, a significant increase in performance will result from the insertion of a delay between each check of the output fifo pointers at label CHK2:. This is because the 8088 on the PGC is put on hold each time the output fifo is accessed by the host for reading/writing data or pointers. A tight loop, checking for available space in the output fifo places a large burden on drawing functions. We have found that about 50 microseconds of delay is about right for most applications.

Vermont Microsystems, Inc. markets its own graphics controller, the VM-8820, which is a two-card set for the PC/AT: it is totally compatible with the IBM PGC. The complete MS-DOS device driver, shipped free with each VM-8820, is also available separately through VMI by calling 802/655-3800.

John M. Cushman
Vermont Microsystems, Inc.
Winooski, VT

AN EQUAL CRUNCHER

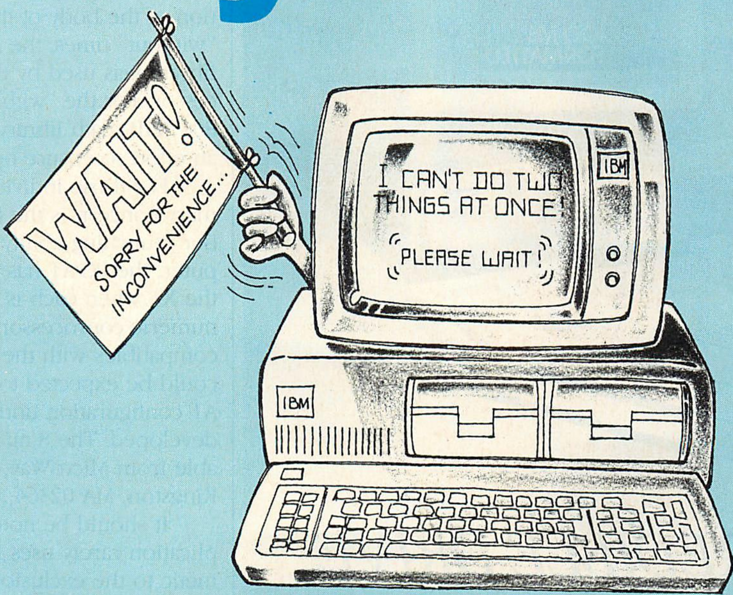
Reviews of the enhanced speed and power of the PC/AT over the PC/XT have shown a three-fold increase for processor-bound tasks and a two- to four-fold improvement in I/O tasks accessing the hard disk. As far as speed is concerned, readers may have the impression that PC/AT time equals PC/XT time divided by 3. This equation is largely true (see "Vroom! Performance Benchmarks for the PC/AT," Susan Glinert-Cole, December 1984, p. 108).

But the three-fold speed increment rule does not hold for computationally intensive tasks using floating-point arithmetic. The 8087 numeric coprocessor in the PC/XT (running at 5 mHz) has roughly the same performance characteristics as the 80287 math coprocessor in the PC/AT (also at 5 mHz). Consider Elite Corporation's NS16032 Consultant Benchmark, a FORTRAN program that begins by initializing a 100-element floating-point array A to the squares of the first 100 integers, then computes 1,000 iterations of the function

$$A(j) = A(j) + 0.5 * (A(j\text{-}1) - 2 * A(j) + A(j+1))$$

for $j = 2, .. 99$. A 512KB PC/XT running DOS 2.0 squared off against a 512KB PC/AT running DOS 3.0 in the test. Without a coprocessor, the XT took 191.4 seconds, the AT only 63.1, for an XT/AT ratio of 3.03. With the coprocessor, the XT took 24.8 seconds, the AT 23.7, for a ratio of 1.05.

Tired of waiting for your PC?



Special Offer Only

$49.95*

This SPECIAL PRICE OFFER absolutely ends on December 31, 1985.

End the wait!... DoubleDOS

Multi-tasking, concurrent processing using today's software and your PC, XT, or AT system!

No matter how good your IBM PC is, you still have to wait while it processes and calculates for a long report, spread sheet, payroll, or whatever. This waiting is costing you time and money.

Now, DoubleDOS lets your PC actually run programs concurrently!

Think of the ways DoubleDOS will save you time and money:

- BUSINESS. Run a spreadsheet, and at the same time, print the payroll checks or accounting reports.
- WORD PROCESSING. Work on another document while printing or spell checking previous work.
- PROGRAMMING. Compile programs while starting new ones immediately.
- COMMUNICATIONS. Receive or send information while you use the PC to do other tasks. DoubleDOS has endless applications!

Order Toll-Free:
1-800-272-9900

CIRCLE NO. 179 ON READER SERVICE CARD

Listen to what users are saying about DoubleDOS:

- "It let's me get more done. Lotus and dBase run together like they were made that way and it only takes 30K in RAM!"
- "Great product! It is solid, painless, and bulletproof!"
- "Simply the best program I've ever had."
- "I'm thoroughly impressed. Fantastic . . . great value-to-cost, too!"
- "Excellent product at an excellent price (hard to find today)."
- "Great stuff! I use it to do compiles and links while editing. Saves a *lot* of time!"
- "Your programming staff deserves a standing ovation for their efforts and results!"
- "I can't believe how fast it works, rarely notice slowdown when using 2 programs."

*Plus $5.00 Shipping & Handling.

SOFTLOGIC SOLUTIONS Satisfaction Guaranteed!

Creators of the Disk Optimizer

530 Chestnut St., Manchester, NH 03101
1-800-272-9900. In NH, call 627-9900.

DoubleDOS works with PC-DOS and your existing software. Also supports many compatibles using MS-DOS.

Version 3.2 of the Microsoft FORTRAN compiler was used with standard four-byte INTEGER and REAL precision. Time duration was estimated by internal program calls to a clock routine that returned the DOS time of day immediately before and after the execution of the body of the code. For the "without" times, the ALTMATH.LIB math library was used by the numeric coprocessor. For the "with" times, the 8087.LIB math library was used, again in an effort to ensure optimal results.

For those individuals whose primary concern is the floating-point number crunching ability of a microcomputer, the PC/AT has little to offer over the XT when each is equipped with the numeric coprocessor. In fact, PC/XT compatibles with the new 8-mHz 8087 could be expected to outperform the AT configuration until faster 80287s are developed. The 8-mHz 8087s are available from MicroWay, P. O. Box 79, Kingston, MA 02364; 617/746-7341.

It should be noted that since an application rarely uses floating-point arithmetic to the exclusion of integer operations, the PC/AT would be expected to show its superiority over the PC/XT to the extent the application is not dominated by floating-point computations.

By the way, the current availability of an array processor is rated at a true mainframe capability of 8 million instructions per second. This will be of interest to PC/XT and PC/AT number crunchers. The array processor system board is produced by Marinco Computer Products, Inc., 3878-A Ruffin Road, San Diego, CA 92123; 619/587-0461.

Thomas J. Lacki, Ph.D.
Bryn Mawr, PA

MicroWay also sells a PC/XT system board with a 10-mHz 8087. Marinco reports that its PC/XT array processor ran a floating-point benchmark 40 times faster than the 8087 and that its PC/AT processor ran the benchmark 49 times faster than the 80287.

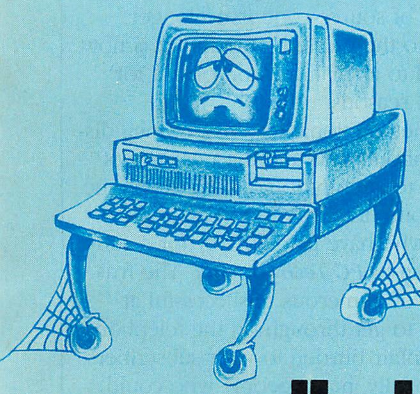
—DB

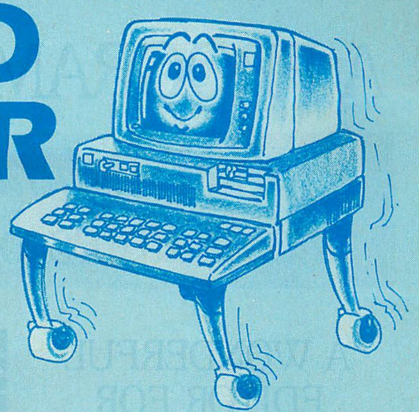
THE PRINTED CODE

I am writing to object to a trend occurring in *PC Tech Journal* and several other computer journals—the policy of not printing source code, but making it availabe for downloading via modems. I believe this new policy violates one of the purposes of this journal.

A very important function of *PC Tech Journal* is to teach by example. Articles on programming must have examples of working code to be maxi-

SHIFT YOUR HARD DISK BACK INTO HIGH GEAR





with

disk optimizer

ONLY

$49.95*

Special Introductory Price

Tunes Your Disk Files For Highest Performance

The Disk Optimizer organizes your disk to make your PC run faster. You'll discover faster program loading, faster file loading and sorting, faster file creation, faster backups to diskette. And the disk Optimizer reduces the chances of error during your disk I/O.

BEFORE DISK OPTIMIZING

A	CC	AA	DDD	D	E	BB	F	FF	GG	AAA	H	BB	JJ	JJJ	AAA	BBBB
A	CC	AA	DDD	D	E	BB	F	FF	GG	AAA	H	BB	JJ	JJJ	AAA	BBBB

AFTER DISK OPTIMIZING

AAAAAAAAA	BBBBBBBB	CC	DDDD	E	FFF	GG	H	JJJJJ
AAAAAAAAA	BBBBBBBB	CC	DDDD	E	FFF	GG	H	JJJJJ

DOS IS SLOWING YOU DOWN!

Whether you realize it or not, the more you use your hard disk, the slower it is likely to react. This is because the operating system (PC-DOS/MS-DOS) stores things (programs, overlays, batches, etc.) randomly, wherever unused disk space is available. Your operating system doesn't keep pieces of the same file together. This means slower access time. Lost speed. Lost productivity.

TURN THE THROTTLE BACK UP!

Now, the Disk Optimizer puts the zip back into your hard disk! The Disk Optimizer collects all the scattered pieces of your files and re-stores them together in neat, well-organized areas on your disk. The result: immediate performance improvement . . . faster loading programs (and their overlays) . . . faster sorting . . . faster loading and storing spreadsheets or document files.

BONUS EXTRAS — at no extra cost

- **DISK ANALYZER**
  Displays a visual performance analysis . . . shown in percentages . . . for any disk or diskette.
- **PERSONAL FILE SECURITY**
  Password protect any file or program.
- **FILE PEEKER**
  Now you can visually examine any file on your system.

*Plus $5.00 Shipping & Handling

$49.95*

Order Toll-Free:
1-800-272-9900

SOFTLOGIC
SOLUTIONS
Creators of DoubleDOS
530 Chestnut Street, Manchester, NH 03101
1-800-272-9900. In NH, call 627-9900
Disk Optimizer works with IBM PC's and true compatibles.

mally effective in teaching. Not printing source code, but merely making it available via modem, severely compromises the important teaching goal of a journal as good as *PC Tech Journal.* The distribution of source code is no longer equal to the printed journal, but is now limited to a small fraction of that for several reasons.

The additional expense of long distance phone calls and the nuisance of having to attempt to get through on the telephone lines are unnecessary for those who have paid already for a subscription to *PC Tech Journal.* The frustration of numerous unsuccessful attempts to get through on the telephone is an unfair burden to your subscribers. Additionally, many people who could benefit greatly from the source code don't have modems. It is unfair to exclude them from obtaining the source code for useful programs.

The main argument given for not printing the source code is to increase the number of articles printed. This is true, but I believe it only increases the quantity of articles printed while decreasing their quality. No scientific paper is considered valid unless a materials and methods section is included in the article; source code is equally important to computer science articles.

For all of the above reasons, I believe this new policy is detrimental to the readership of *PC Tech Journal* and should be eliminated. The extra pages necessary to print the code are a small price to pay to maintain the high standards of *PC Tech Journal* as well as to maximize its usefulness to the greatest number of people.

In particular I was interested in all of the I/O routines that were described in "Improving Turbo's I/O," Cole Brecheen, September 1985, p. 104.

Robert Solomon, M.D.
Forest Hills, NY

This was a tough call for us. The size of the "Improving Turbo's I/O" article with listings was prohibitive. We had to choose between running those listings or running another article. Because not all of our readers are interested in the subject of Turbo Pascal's I/O, we decided in favor of balance.

This was an exception to our normal policy and not a trend toward fewer listings. As a rule, we try to run all listings that pertain to an article. For those cases in which the listings are too big, we will run the most important parts and make the remainder available on PCTECHline. Furthermore,

Back up 10 megabytes in 8 minutes! (or less)

with Fastback™

If you've ever lost valuable data, you know backing up your hard disk is a *MUST.* For your business, this necessity can become an expensive nightmare. Until now. Fastback™(Version 5.0) is the hard disk backup (and restore) software utility that eliminates the need for expensive and time consuming tape backup systems. It's so fast you can back up a full 10-megabyte hard disk on standard 5¼″ floppies in less than 8 minutes or 10 megabytes of data on the IBM™PC-AT in less than 4 minutes! This can mean a 95% savings of time and money. And it's easy to use.

Fastback is fully self-contained and functions with any make or size hard disk. It works with PC-DOS™, or MS-DOS™version 2.0 or higher and requires no additional hardware to take up valuable expansion slots in your IBM PC/XT/AT or compatible. FASTBACK USES ADVANCED *ERROR CORRECTION* TECHNIQUES TO RECOVER DATA FROM DAMAGED DISKETTES. So if your disk is damaged after the backup procedure, Fastback (Version 5.0) will still be able to recover the data.

Fastback allows rapid lookup of archived data and prevents accidental overwriting. Fastback is completely file oriented, not an image mode backup.

Suggested Retail Price: **$179.**

INTRODUCTORY OFFER

Yes, send me________Fastbacks at the Special Introductory price of $149 each. That's $30 off the regular retail price of $179.

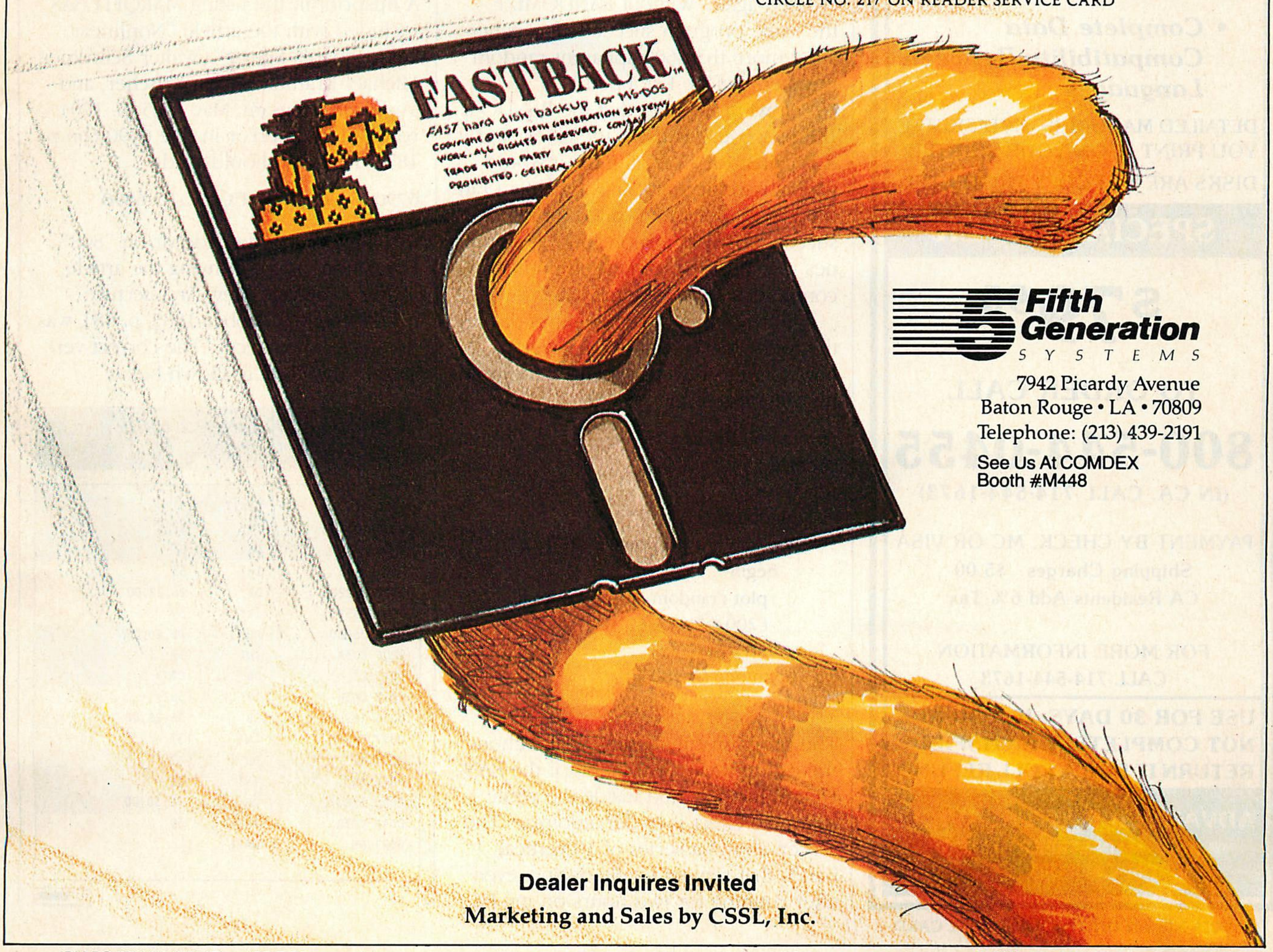
I am enclosing: ☐ BankDraft ☐ Check **$30 OFF**

Name:
Address:
City:
State: Zip:
Telephone (____)

These prices include shipping to all U.S. cities. All foreign orders add $10 per product ordered. COD's and purchase orders WILL NOT be accepted by Fifth Generation Systems. Outside USA: payable in U.S. dollars drawn on a U.S. bank. FASTBACK is also available at better dealers nationwide. Call (213) 439-2191 for the dealer nearest you.

Offer Expires December 31, 1985

CIRCLE NO. 217 ON READER SERVICE CARD



Fifth Generation SYSTEMS

7942 Picardy Avenue
Baton Rouge • LA • 70809
Telephone: (213) 439-2191

See Us At COMDEX
Booth #M448

Dealer Inquires Invited

Marketing and Sales by CSSL, Inc.

IBM-PC, IBM-XT, IBM-AT and PC-DOS are registered trademarks of International Business Machines, Inc. MS-DOS is a registered trademark of Microsoft, Inc. Fastback is a registered trademark of Fifth Generation Systems.

PCTECHline sometimes includes files that could not be published, such as the executable (.COM, .EXE) versions of listings that are complete programs.

PCTECHline is not an excuse to run fewer listings. It is an added-value service to our subscribers. We are committed to increasing its availability; we ask your patience as we add phone lines and equipment.

—*WF*

AT RANDOM

Like Mr. Laird ("Letters," August 1985, p. 16), I wondered about the Turbo Pascal RANDOMIZE procedure. I have Turbo 3.0, so I assumed that the problem had been fixed when I ran KEYGEN without an apparent problem.

Further investigation, however, reveals a different situation. A simple Turbo program to repeatedly print a random number appears to do so when RANDOMIZE is not used; the program, however, appears to generate a predetermined number when RANDOMIZE is used. Using RANDOMIZE, the same number appeared 80 times in succession and than increased by 1. The new number then appeared 80 times and increased again. Without RANDOMIZE, the same program succeeded in generating more than 600 apparent random numbers before I stopped it.

It would appear that the RANDOMIZE procedure is using the clock, and if it is called on a casual basis it would probably achieve its aim of giving a random seed. But when called rapidly in succession, as in a short loop, it continues to repeat the same seed until the computer's clock changes.

For readers with a graphics display, the following Turbo Pascal program may serve to reassure them of the true randomness of the system:

```
program randscrn;
 begin
   hires;
   randomize;
     while not keypressed do
     begin
      plot (random (640), random
       (200), 7;
     end
   textmode (bw80);
 end.
```

The program runs just as well without the call to RANDOMIZE, but if the RANDOMIZE is inserted inside the while loop, the effect is to derandomize. The screen shows a series of 10 vertical bands that remain totally distinct even after running for two hours on my PC.

Users of IBM BASIC should try the following program:

```
10 SCREEN 2
20 X = RND * 640
30 Y = RND * 200
40 PSET (X,Y)
50 GOTO 30
```

RANDOMIZE makes no difference here either, and entering different seed values doesn't shift the pattern as one might expect.

William T. Marchant
Halifax, Nova Scotia

SUPPORTED BY AUTOCAD

An omission in your "Power Graphics" article (Thomas V. Hoffman, July 1985, p. 56) should be cleared up.

The author mentioned that AutoCAD supports the Vectrix VX/PC controller, but failed to mention that it also supports the IBM Professional Graphics Controller (PGC). Support for this product was announced in March 1985.

Sandra Boulton
Autodesk, Inc.
Sausalito, CA

ERRATA

A misprint in the listing MARQFIT.BAS included with the article "Nonlinear Least-squares Fitting" (Walter Schreiner, Michael Kramer, Simon Krischer, and Yedidyah Langsam, May 1985, p. 170) resulted in an error in line 6790 (on p. 187). The line should read:

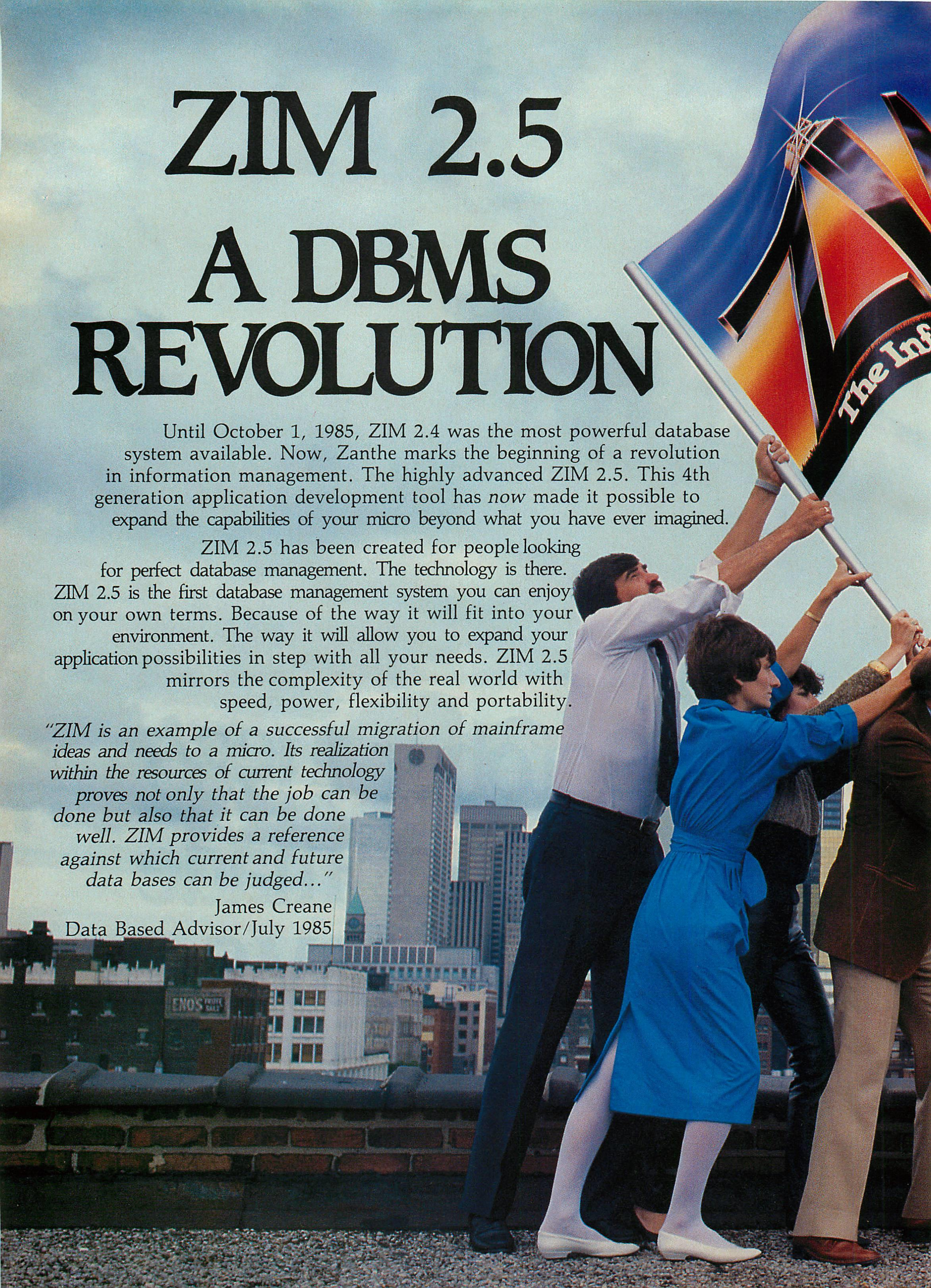
```
6790 IF NO = NFIT%, GOTO 6890
```

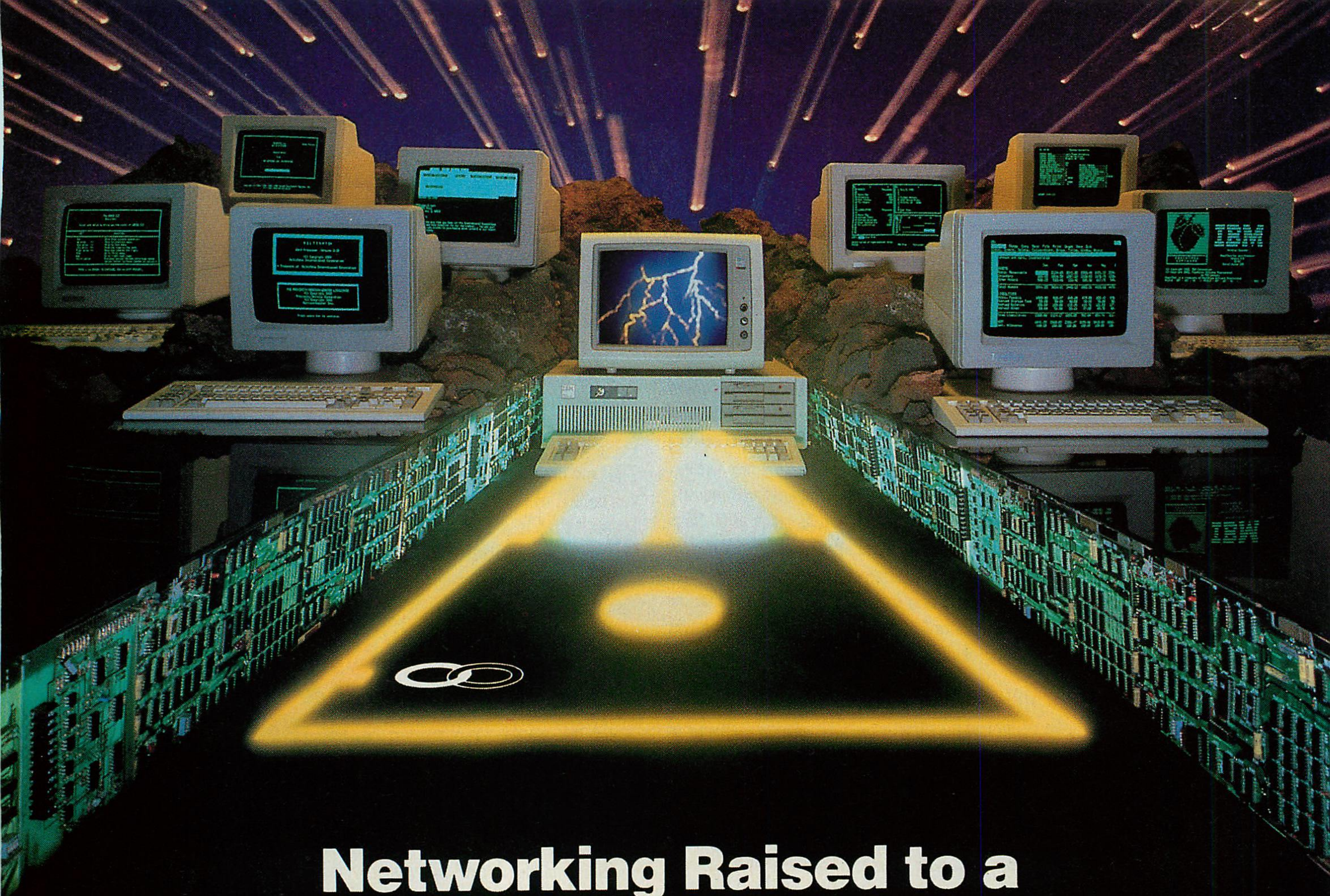
Figure 3 of the sidebar on "8087 Emulation" accompanying the article, "Same Language, New Architecture" (Ted Mirecki, October 1985, p. 48), was presented incorrectly. The correct version of the code is shown below.

FIGURE 3: *Linked with Emulator Library*

```
0000 DBE3      FINIT
0002 CD37      INT    37
0004 E3        DB     E3
0005 CD35      INT    35
0007 062200    DB     06,22,00
000A CD39      INT    39
000C 062600    DB     06,26,00
000F CD3A      INT    3A
0011 C1        DB     C1
0012 CD3B      INT    3B
0014 162E00    DB     16,2E,00
0017 CD37      INT    37
0019 17        DB     17
001A CD39      INT    39
001C 3E3000    DB     3E,30,00
001F CD3D      INT    3D
0021 9B        WAIT
```

On October 1, 1985 one product made all other DBMS obsolete.


ZIM 2.5
A DBMS REVOLUTION
Until October 1, 1985, ZIM 2.4 was the most powerful database system available. Now, Zanthe marks the beginning of a revolution in information management. The highly advanced ZIM 2.5. This 4th generation application development tool has *now* made it possible to expand the capabilities of your micro beyond what you have ever imagined.
ZIM 2.5 has been created for people looking for perfect database management. The technology is there. ZIM 2.5 is the first database management system you can enjoy on your own terms. Because of the way it will fit into your environment. The way it will allow you to expand your application possibilities in step with all your needs. ZIM 2.5 mirrors the complexity of the real world with speed, power, flexibility and portability.
"ZIM is an example of a successful migration of mainframe ideas and needs to a micro. Its realization within the resources of current technology proves not only that the job can be done but also that it can be done well. ZIM provides a reference against which current and future data bases can be judged..."
James Creane
Data Based Advisor/July 1985



Networking Raised to a Greater Power

Advanced Technology. With it, IBM tripled the speed of the PC and increased its memory capacity five-fold. Nowhere is this increase in computing power more important than in networking situations. If the AT's technological advances have prompted you to look into a multi-user network, you owe it to yourself to take a closer look at MultiLink Advanced™ . . . a unique multi-tasking, multi-user networking system that runs programs under PC-DOS 3.0.

Eight Workstations for the Price of an AT. MultiLink Advanced™ represents the next generation in networking systems for IBM microcomputers. The system enables terminals, connected to a single AT, to emulate IBM-PC's having up to 480K of RAM (The PC-Shadow™ terminal, shown above, even has a PC look-alike, as well as work-alike, keyboard and display).

This means that instead of spending $3,000 per workstation for a PC with a Kilobuck "Network Interface Board," you can use inexpensive terminals . . . eight of which cost less than an IBM AT. Even if you need only one workstation connected to your AT, you'll realize significant savings.

MultiLink Advanced™ . . . Instant Access to All of Your Resources. Central to most multi-user situations is the need to coordinate a variety of printers. With what's been described by *PC-Tech Journal* as ". . . by far, the best print spooler for the IBM PC," MultiLink Advanced™ gives users the option to print either at their workstations, or at a central location. In addition, programs and files can be shared by multiple users locally or through use of a modem. Just think of it . . . having remote access to an AT with a lightweight terminal/modem.

Although designed to take advantage of the AT, MultiLink Advanced™ runs on all versions of PC-DOS, except 1.0, and certain implementations of MS-DOS. A wide range of leading programs are supported which include WordStar, dBASE III, Multimate, and Lotus 1-2-3.

Get the Advanced Story Today. Call The Software Link Today for complete details and the dealer nearest you. MultiLink Advanced™ is immediately available at the suggested retail price of $495 and comes with a money-back guarantee. VISA, MC, AMEX accepted.

MultiLink™
ADVANCED

See us at Booth #M818
COMDEX/Fall '85
MGM Grand Hotel

THE SOFTWARE LINK, INC.

8601 Dunwoody Place, Suite 336, Atlanta, GA 30338 Telex 4996147 SWLINK
CALL: 404/998-0700

Dealer Inquiries Invited

IBM, PC, AT, & PC-DOS are trademarks of IBM Corp. MS-DOS, WordStar, dBASE III, Lotus 1-2-3, and Multimate are trademarks of Microsoft Corp., MicroPro, Ashton-Tate, Lotus Development Corp., & Multimate International, respectively.

MultiLink Advanced™ is a trademark of The Software Link, Inc.

CIRCLE NO. 143 ON READER SERVICE CARD

JEFF DUNTEMANN

Successfully Seeking Interrupts

IBM's PCWatch helps in developing systems software by treating interrupts as individual machine events.

The IBM PC's systems-level software is a web woven of hardware and software interrupts. Interrupts mediate the transfer of information to and from hardware peripherals. Software interrupts allow programs to call routines whose locations may be changeable or unknown at compile time. Writing and debugging system software requires good tools and a great deal of patience.

Missing from the debugger model of machine operation is the concept of the interrupt as an individual machine event that may be observed, trapped for, and single-stepped. (Most debugger-type tools are instruction-oriented.) IBM has filled that gap with part of its Personally Developed Software series. *PC Tech Journal* presents PCWatch as December's Product of the Month.

PCWatch is not itself a debugger. It provides a means of selecting interrupts to be observed and displaying information about interrupt events as they happen in the PC. A well-behaved interface to resident debuggers, such as DOS DEBUG and IBM's Resident Debug Tool, allows PCWatch to drop into the debugger either before a selected interrupt begins executing or after it has executed but immediately before it returns.

At minimum, PCWatch displays the name of the event (for example, INT 10 Write TTY), an event count since PCWatch was invoked, and the state of all 8086 registers prior to interrupt entry. Registers may optionally be displayed on interrupt exit. Some events that use other types of parameters, such as ASCIIZ strings and FCBs, will display those as well. PCWatch should be used on a two-monitor system so full-screen applications may be monitored without disruption by PCWatch output. A split-screen display option for single-monitor systems is available. Output may be directed to a printer.

PCWatch output may be suspended and restarted at any time. Its internal monitoring functions continue even if the display is suspended. The monitoring functions may be turned off, but they may not then be turned on again until the user has rerun PCWatch from the command line.

This simple concept is made powerful by set-up flexibility. Interrupt events are listed on the menus both singly and divided into functional groups. BIOS interrupts are further divided into subfunctions. Any event or group of events may be included in or excluded from display. This allows the display of an entire related group of events such as VIDEO *except* for selected events such as Write TTY. A NOISE category may be excluded to prevent timer ticks and other regular events from overwhelming the display. Nested events (interrupts from within interrupts) may be either included or excluded as the user desires.

PRODUCT NAME
PCWatch

COMPANY
IBM
Personally Developed Software

ADDRESS
P.O. Box 3280
Wallingford, CT 06494

TELEPHONE
800/IBM-PCSW

PRICE
$49.95

As with all programs that watch what an application is doing, PCWatch exacts a performance penalty. It works by redirecting interrupt vectors. When no interrupt events occur, performance degrades by 10 percent or less. Beyond that, the degree of degradation depends directly on the number of interrupt events that occur and the amount of information being displayed.

One important factor in the selection of PCWatch as Product of the Month is its overall excellence of design. The bounce-bar set-up menu screen is rationally arranged and remarkably easy to understand. Options may be selected with a minimum number of keystrokes. The use of color adds to clarity rather than distraction.

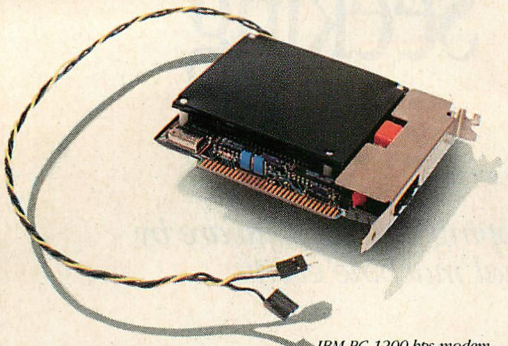
With as many variations in set-up as are possible with PCWatch, the ability to load a set-up without having to enter it through the menu each time is important. PCWatch set-up is table-driven by way of an ASCII file in which one line governs each configurable option. The file contains comment lines and is easily readable. Changes are made to the table file via any text editor, and then a compiler program included with PCWatch combines a binary translation of the table with the PCWatch code itself, producing an entirely new version. This allows a developer to produce a broad-strokes custom version of PCWatch for a particular development project, and then fine-tune the custom version from the menu.

The configuration table allows options not available from the menu, including the ability to display the stack, and arbitrary memory blocks on entry to or exit from an interrupt event.

Aside from its obvious uses in developing software, PCWatch is helpful in customizing commercial packages to handle different peripherals. It would be easy to locate the mode-set VIDEO call in a program, allowing a patch to bring up a different video mode for a nonstandard display adapter.

PCWatch's cost has been kept down by bare-bones packaging and the distribution of the documentation as a DCA text file on the program disk. At $49.95, PCWatch is very inexpensive for what it accomplishes, and it belongs in every serious programmer's toolkit.

Hardware, software, and other developments for the IBM PC family



IBM PC 1200-bps modem

IBM 5841 modem

FROM IBM

The **IBM Quietwriter** model 2 offers all-points-addressable, high-resolution graphics in addition to all the features and functions of the model 1. This new model also includes a performance switch that allows users to choose the best print mode for a particular application and an increased print buffer to store graphing commands in the printer and allow for multitasking. An upgrade kit is available to allow users to add these graphics capabilities to the model 1. **IBM Quiet**, a new noncorrecting ribbon for Quietwriter that produces dark, indelible images for printing on a broader assortment of papers, also has been introduced. The ink from this ribbon cannot be erased without detection. Quietwriter model 2, $1,595; upgrade kit, $350; Quiet ribbon, $12.

IBM Quietwriter

The IBM Personal Computer **1200-bps modem** is mounted on a circuit card for easy installation in the PC, PC/XT, PC/AT, and Portable PC. The **IBM 5841** 1200-bps modem is a stand-alone unit for use with PCs and IBM 3161/63 ASCII display stations. Both modems provide asynchronous data transmission; the 5841 also allows synchronous transmission. They feature automatic speed detection and line equalization; automatic detection of dial tones, busy, ringback, and attention signals; automatic or manual dial; and automatic redial. Internal modem, $499; IBM 5841, $609.

Under a new volume maintenance amendment, IBM customers can receive a 15-percent discount on service for 150 to 499 qualifying IBM PC system units; a 20-percent discount for service of 500 to 999 units; and a 25-percent discount for service of 1,000 or more.

Also from IBM comes **Interactive System Productivity Facility/Program Development Facility Personal Computer Editor** (ISPF/PDF PC Editor), a PC version of the ISPF/PDF that runs on IBM mainframes. The PC Editor, a productivity aid, enables programmers familiar with ISPF/PDF to develop mainframe computer applications on the PC. This, in turn, can result in reduced dependency on the host computer and better response time. ISPF/PDF PC Editor allows programmers to delete, repeat, move, or modify sections or lines of code as well as elements within a line by typing the appropriate command rather than by making manual line-by-line changes. **EZ-VU Runtime Facility**, a program that provides conversational interface between the PC user and an applications program, is a prerequisite for the PC Editor. ISPF/PDF PC Editor, $150; EZ-VU Runtime Facility, $55.
IBM Corporation, Information Systems Group, 900 King Street, Rye Brook, NY 10573; Contact the local IBM dealer
CIRCLE 301 ON READER SERVICE CARD

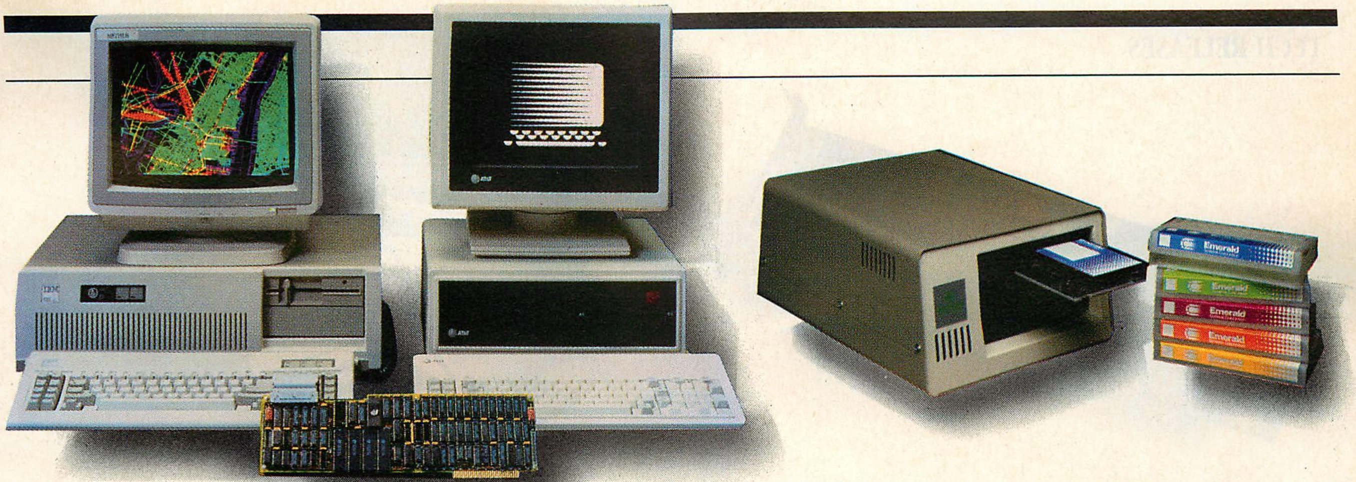
HARDWARE

A new hardware/software package that facilitates bidirectional file transfer between the PC and the IBM System 3x has been introduced by **AST Research, Inc.** The **AST-VDI**, used in conjunction with AST's 5251/11, provides a direct interface to IBM's file support utility. By employing a virtual diskette interface technique that simulates a PC diskette drive set up in the System 3x fixed disk, AST-VDI speeds up file transfer.

AST also has introduced a PC-compatible video adapter as an addition to its line of Model 11 micro-to-System 34/6/8 terminal emulation packages. The **AST-5250/Display**, when used in conjunction with the AST-5251/11 terminal emulation package, enables PCs equipped with a standard IBM monochrome monitor to support all of the IBM 5251 model 11 character and field display attributes, some of which cannot be displayed using the standard IBM video adapter. The adapter includes a parallel printer port, a Hercules-compatible graphics mode, and is compatible with bit-mapped graphics and text and windowing applications software. $495.
AST Research, Inc., 2121 Alton Avenue, Irvine, CA 92714; 714/863-1333
CIRCLE 306 ON READER SERVICE CARD

Bit-slice graphics is now available for the IBM PC/AT with the **T4 Color Graphics Controller** from **Microfield Graphics, Inc.** This high-performance graphics engine provides 1,024-by-800 pixel resolution and writing speeds of more than 1 million pixels per second. T4 architecture is based on low-energy, CMOS bit-slice gate arrays that contribute both to performance levels and IBM compatibility; it is completely microprogrammable. The **T4 Development Toolkit** allows OEMs to use the T4 to create proprietary graphics applications rapidly. T4 Color Graphics Controller, $3,200; Development Toolkit, $100.

To provide an interface between the T4 Color Graphics Controller and the AT, Microfield also announced its proprietary version of a VDI: the **Microfield Virtual Device Interface** (MVDI) allows users to draw device-independent graphics and makes already developed graphics applications more easily portable from other computers. $16,000.
Microfield Graphics, 8285 S.W. Nimbus Avenue, Suite 161, Beaverton, OR 97005; 503/626-9393
CIRCLE 310 ON READER SERVICE CARD



Omega PC by Metheus Corporation

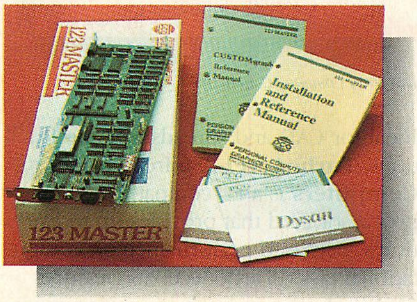
XENIX tape backup by Emerald Systems

Omega PC is a high-performance, 1,024-by-1,024 pixel color graphics controller for the PC/AT and compatibles. Produced by **Metheus Corporation**, it provides a screen resolution of 1,024 by 768 pixels. Its four bit planes permit simultaneous display of 16 colors from a palette of 4,096; 60-Hz noninterlaced refresh provides flicker-free display. The controller requires two expansion slots in the host PC. $2,500.
Metheus Corporation, 5510 N.E. Elam Young Parkway, P.O. Box 1049, Hillsboro, OR 97124; 503/640-8000
CIRCLE 307 ON READER SERVICE CARD

IOMEGA **Corporation** introduced four new **Bernoulli Box** subsystems, including 10MB single-drive and 20MB dual-drive versions with average access times of 35 milliseconds, and 20MB single-drive and 40MB dual-drive versions with access times of 38 milliseconds. Their data transfer rate is 1.13MB per second. The half-height boxes combine Winchester performance with removable-media features to produce a unit that is immune to head crashes and problems of vibration or contamination. Bernoulli 10 (10MB single-drive), $2,695; 10 + 10 (20MB dual-drive), $3,695; 20 (20MB single-drive), $3,295; 20 + 20 (40MB dual-drive), $4,695.
IOMEGA Corporation, 1821 W. 4000 South, Roy, UT 84067; 801/778-1000
CIRCLE 308 ON READER SERVICE CARD

Personal Computer Graphics Corporation has announced two new graphics boards. The **Photon 96** is a high-speed, high-resolution graphics card for the PC/XT and PC/AT that has applications in CAD/CAM, medical imaging, advanced realtime simulation, graphic arts, and slide presentation. This board supports automatic line, vector, circle, rectangle, arc, space fill, and programmable shape functions. The user can obtain speeds up to 10 nanoseconds per pixel, as well as programmable resolution from 640 by 200 to 1024 by 1024. Sixteen colors and a light pen interface are provided. $1,695.

The company also announced **123 MASTER**, a graphics card that uses an IBM monochrome or color display monitor to run graphics software such as Lotus 1-2-3 and Lotus Symphony. 123 MASTER gives the user the ability to perform enhancements to any graphics image while the image is running. In addition, it provides the user access to graphics with only one keystroke. Monochrome card, $499; color card, $695.
Personal Computer Graphics Corporation, 909 South Broadway, P.O. Box 6298, Los Angeles, CA 90055; 213/612-0879
CIRCLE 304 ON READER SERVICE CARD

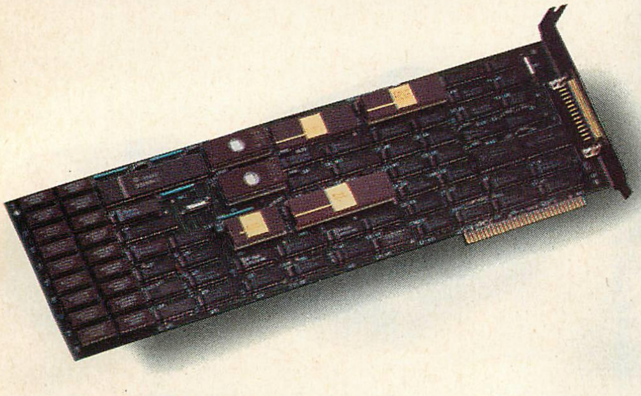


By Personal Computer Graphics Corporation

An ultra-high speed interface that transfers bit-mapped images directly from the computer's RAM to the laser printer mechanism has been announced by **Tall Tree Systems. JLASERPRINTER** bypasses the usual RS-232 serial interface, the on-board printer buffer, and most of the control circuitry in conventional laser printers. It is a daughterboard that fits on any JRAM 2MB board. It supports installable device drivers for Lotus/Intel expanded memory (16KB blocks), the Tall Tree Systems bankswitching standard (64KB blocks), the JETDRIVE RAM disk package for multi-MB RAM disks, and JSPOOL, which can use memory beyond 640KB for a conventional print-spooler buffer. JLASERPRINTER, $400; with JRAM-2 memory-board, 1,280KB RAM, $849; with JRAM-3 (Lotus/Intel/Microsoft memory), $999; with JRAM-AT3 (16-bit Lotus/Intel/Microsoft memory), $1,049.
Tall Tree Systems, 1120 San Antonio Road, Palo Alto, CA 94303; 415/964-1980
CIRCLE 309 ON READER SERVICE CARD

The first **XENIX tape backup subsystem** for the PC/AT has been introduced by **Emerald Systems Corporation**. The new tape subsystem uses the standard XENIX backup utilities to store up to 60MB on a one-fourth-inch cartridge. It supports all of Emerald's XENIX hard drives (36 to 140MB) as well as the IBM 20MB drive. Internal system, $2,050; external system, $2,250.
Emerald Systems Corporation, 4757 Morena Blvd., San Diego, CA 92117; 619/270-1994
CIRCLE 311 ON READER SERVICE CARD

A multifunction server called **PLAN 5000** from **Nestar Systems, Inc.** supports 25 to 50 workstations. The base PLAN 5000 provides 275MB or 550MB of formatted disk capacity. Storage capacity may be expanded to 825MB or 1.1GB. Unformatted storage capacity equivalents are 344, 688, 1,032 and 1,376 millions of bytes, respectively. A 60MB streaming tape drive for backup is standard and a second 60MB is optional. Additional features include Nestar's Shadow, which provides fault tolerance via a secondary file server for backup, an integrated print server coprocessor capable of driving three printers simultaneously, the PLANpak starter library of network-licensed multiuser applications software, and the server console. PLAN 5000 with formatted capacity: 275MB, $25,000; 550MB, $35,000; 825MB, $45,000; 1,100MB, $55,000; second 60MB streaming tape backup, $2,000.
Nestar Systems, Inc., 2585 E. Bayshore, Palo Alto, CA 94303; 415/493-2223
CIRCLE 315 ON READER SERVICE CARD



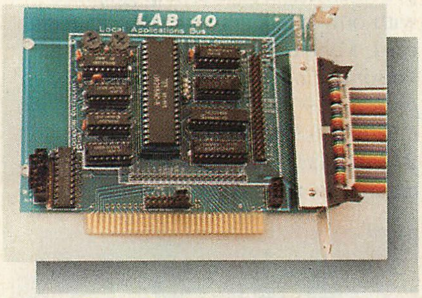
Quad3270 by Quadram

Addition to Priam's InnerSpace family

Language Resources, Inc. has introduced a 32-bit software development system for the Motorola 68020 called **PC-020**. This upgrade package includes a 68020 plug-in CPU board with a 68881 math coprocessor and 1MB of RAM, a symbolic debugger, a Motorola compatible assembler, compilers, utility software, and documentation. In addition, it provides the complete facilities necessary for engineers to write, assemble, compile, test, and execute 68020 code directly on their PCs. $5,995; compilers for C and Pascal, $895 each.
Language Resources, Inc., 4885 Riverbend Road, Boulder, CO 80301; 303/449-6809
CIRCLE 318 ON READER SERVICE CARD

A high-capacity, 5¼-inch drive for the PC and the diskless PC/XT has been announced by **Priam Corporation**. This addition to the **InnerSpace** family of add-in Winchester disk drives offers users 43MB or 60MB capacities with a data access speed of 30 milliseconds and fits into the same internal drive slot as the standard PC 5¼-inch floppy disk drives. The complete add-in kit includes a disk drive, disk controller, mounting hardware, installation software, instruction manual, and reference guide; it requires no additional interfaces. The drives feature automatic defect management to protect data integrity, a dedicated head landing zone, shock mounts, and automatic head lock. 43MB ID40-PC, $1,998; 60MB ID60-PC (includes required disk controller), $2,298.
Priam Corporation, 20 W. Montague Expressway, San Jose, CA 95134-2085; 408/946-4600
CIRCLE 312 ON READER SERVICE CARD

The Local Applications Bus, **LAB 40**, is a computer-to-peripheral interface and a hardware development system from **Computer Continuum**. LAB 40 can interface directly to microprocessor compatible ICs and hybrids without additional circuitry. Applications include data acquisition, control, test instrumentation, and conventional computer peripherals. The bus is a structured parallel port with 16 data lines, 8 controls, 8 selects, 7 interrupts, and power lines. $200.
Computer Continuum, 75 Southgate Avenue, Suite 6, Daly City, CA 94015; 415/755-1978
CIRCLE 303 ON READER SERVICE CARD



LAB 40 by Computer Continuum

Ideal for CAD and spreadsheet applications, **TurboAccel-286** from **Earth Computers** is a PC-compatible 80286 processor card that provides up to five times increased performance on computational tasks. TurboAccel-286 features 512KB of memory with expansion capability, provisions for an optional 80287 math coprocessor, and a switch to run the original 8088 microprocessor. The board is compatible with most operating systems because it uses the El-286-88 chip to emulate the 8088 signals and functions directly. $995.
Earth Computers, P.O. Box 8067, Fountain Valley, CA 92728; 714/964-5784
CIRCLE 302 ON READER SERVICE CARD

Quad3270 Gateway from **Quadram Corporation** provides a communications link between LAN-based PCs and SNA (systems network architecture) hosts. With 512KB of main memory and a 32-bit microprocessor, the Quad3270 Gateway can handle varied communication functions. It supports PU type 2 and LU types 1, 2, and 3 sessions and services, and it can emulate a 3274-51C/61C remote communications controller. Quad3270 provides for 3270 Model 2 display emulation, four-color 3279 emulation, and 3287 printer emulation; it accommodates up to 9600-baud synchronous host communications. 8 LU, $4,529; 16 LU, $4,983; 32 LU, $5,742.
Quadram Corporation, 1009 Mansell Road, Roswell, GA 30076; 404/993-4590
CIRCLE 305 ON READER SERVICE CARD

Half Card 24 is a 2400-baud internal modem for the IBM PC line. Produced by **Ven-Tel Inc.**, this 5¼-inch card uses the industry-standard PC/AT command set and makes upgrading to 2400 baud easy. The Half Card 24 features comprehensive local and remote test modes, call progress reporting, automatic speed selection, and fallback to smooth the transition from higher to lower speed transmissions. Crosstalk XVI, provided with the modem, supports file transfer, data capturing, terminal emulation, and script processing. $695.
Ven-Tel, Inc., 2342 Walsh Avenue, Santa Clara, CA 95051; 408/727-5721
CIRCLE 313 ON READER SERVICE CARD

Anchor Automation, Inc. has added a 300/1200/2500-bps modem to its Signalman family. The **Signalman Lightning 24** operates asynchronously at 2400 bps with an automatic equalizer to ensure clear transmission. Lightning 24 provides fallback to 1200 or 300 bps. External switch settings permit unattended or manual configuration and operation. Features include permanent multiple telephone number storage and autodial, line monitoring, self diagnostics for power up, analog and digital loopback, and battery backup. $599.
Anchor Automation, Inc., 6913 Valjean Avenue, Van Nuys, CA 91406; 818/997-7758
CIRCLE 314 ON READER SERVICE CARD

READ  ONLY

A review of the IBM Personal Computer Family. Vol. 2, No. 3

HARDWARE NEWS

A growing family of PC Printers.

Different jobs demand different printers. IBM has developed a full line of printers for its Personal Computer Family to answer those demands. Two of the most recent—the IBM Proprinter and the IBM Color Jetprinter—deserve special mention.

Both are compatible with the IBM PC Family of Personal Computers as well as with many other leading home and office personal computers. And both offer the traditional high standard of IBM reliability and support.

The IBM Proprinter.

The IBM Proprinter alone can help you complete a variety of different home or office jobs.

It has an unusual feature that will be welcomed by everyone who has spent time switching back and forth from single-sheet to continuous forms printing. A special slot in the front of the Proprinter allows you to print single pages or envelopes quickly and easily without removing the continuous forms paper.

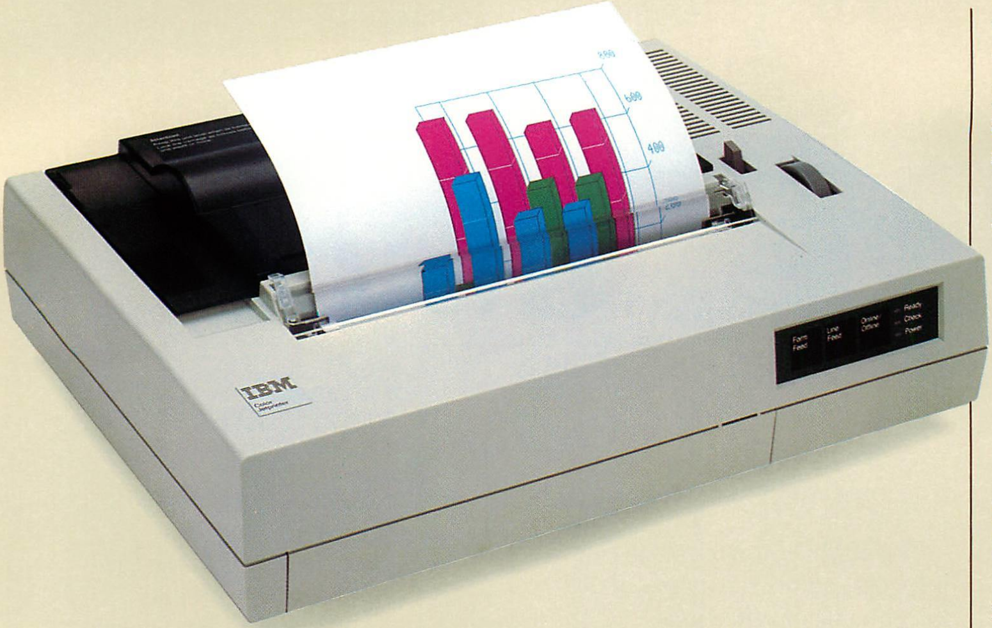
The Proprinter's high-speed (200 cps), dot-matrix technology can make short work of your day-to-day printing tasks like first drafts of letters or reports.

For more finished work, the Proprinter does near-letter-quality printing (at 40 cps) and lets you add emphasis to your text (at 100 cps) with bold print, condensed print and underlining.

It also has all-points-addressable graphics capabilities that can help you produce a complete range of charts and graphs to illustrate your text.

One additional plus: it's very competitively priced.

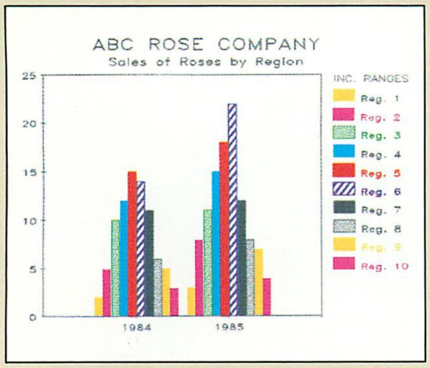
IBM Proprinter has a convenient slot for printing envelopes or single pages.

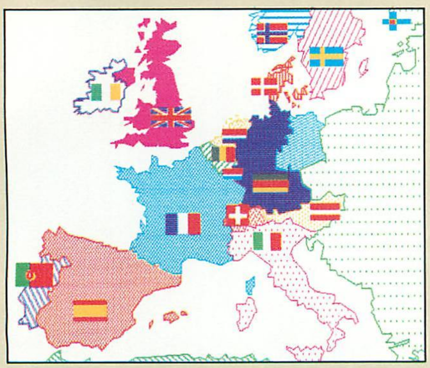


IBM Color Jetprinter quietly utilizes "drop-on-demand" ink jet technology.

The IBM Color Jetprinter.

The IBM Color Jetprinter is an unusually quiet, compact workstation printer that utilizes advanced "drop-on-demand" ink jet technology.





Samples of color graphics and texts produced by IBM Color Jetprinter.

Designed to be used with the IBM PC and many other personal computers, it's supported by a wide selection of software programs.

You can use it to produce draft and near-letter-quality text and high-quality graphics in up to seven colors.

In addition, the Jetprinter allows you to highlight your text with bold and underlined passages done in color or printed against a background color.

The Color Jetprinter also prints graphs, charts, engineering drawings and texts directly onto special Jetprinter transparencies for presentations.

And, thanks to a unique non-clogging ink formula, it performs all these functions very reliably.

FAMILY TIES

Videotex connections.

Videotex provides a key to much of our future shopping, entertainment and business information. Videotex services, using existing telephone lines, already supply travel information, personal shopping options and personalized messaging, banking and investing services.

PC Videotex software from IBM allows you to access these services from your IBM Personal Computer, IBM Personal Computer XT,™ IBM Personal Computer AT® or IBM PC*jr*.™

With only a few simple commands, you can use your IBM PC to receive, display, store and replay the data you choose in high-resolution color graphics and texts.

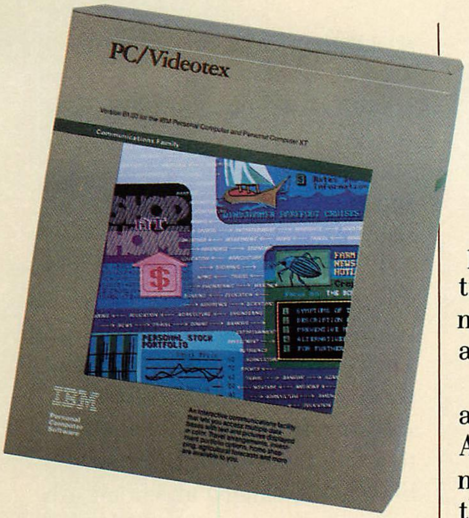
And if you need more than information retrieval, PC Videotex, in certain configurations, can be used to support your own Videotex data base.

WHAT'S THE PROGRAM?

Create your own presentations.

Everyone who has ever put together a visual presentation for a business meeting or other educational purpose can now take heart. With the help of IBM PC Storyboard software, an IBM PC and a color graphics monitor you can turn out professional-looking, captivating presentations complete with color, special effects and eye-catching graphics.

PC Storyboard's four modules—Picture Taker, Picture Maker, Story Editor and Story Teller—make it fast, easy and comparatively inexpensive to create, combine and modify pictures and stories. There's even an animation function that makes it possible for birds to fly, ships to move across water, words to become bigger.



IBM PC Videotex software makes Videotex connections in high-resolution color graphics and text.

Picture Taker lets you store the contents of screen displays from software you regularly work with, such as spreadsheet, business graphics, drawing or word processing programs.

You can use the Picture Maker module to edit and combine the graphics you've saved. Picture Maker also allows you to create completely new pictures containing both text—in a variety of type styles—and graphic figures.

Story Editor helps you organize and edit your pictures into a presentation and add various special effects. The results can be dramatic.

You can then use the Story Teller module to display the results on an IBM PC Color Display, on a TV or with a video projector.

The finished presentation can be copied on diskettes for distribution. You can also produce printed copies with a graphics printer like the IBM Color Jetprinter reviewed in this issue of *Read Only*.

After that, all you need is a director's chair from which to view the finished product.

Expand capabilities with IBM Planning Assistant Solutions, IBM Document Retrieval Assistant and PC Storyboard software.

Information at your fingertips.

A few months ago in this space, we reviewed Office Correspondence Retrieval System (OCRS) software from IBM, which provides a convenient way to keep track of information that otherwise might be filed and never found again.

That same application is now available as a member of the IBM Assistant Series, with an easy-to-use menu system that's consistent with those found in other Assistant Series programs.

Like the original OCRS software, IBM Document Retrieval Assistant makes life much easier for anyone who stores large amounts of information on a fixed disk or who has a library of documents, memos or files stored on diskettes.

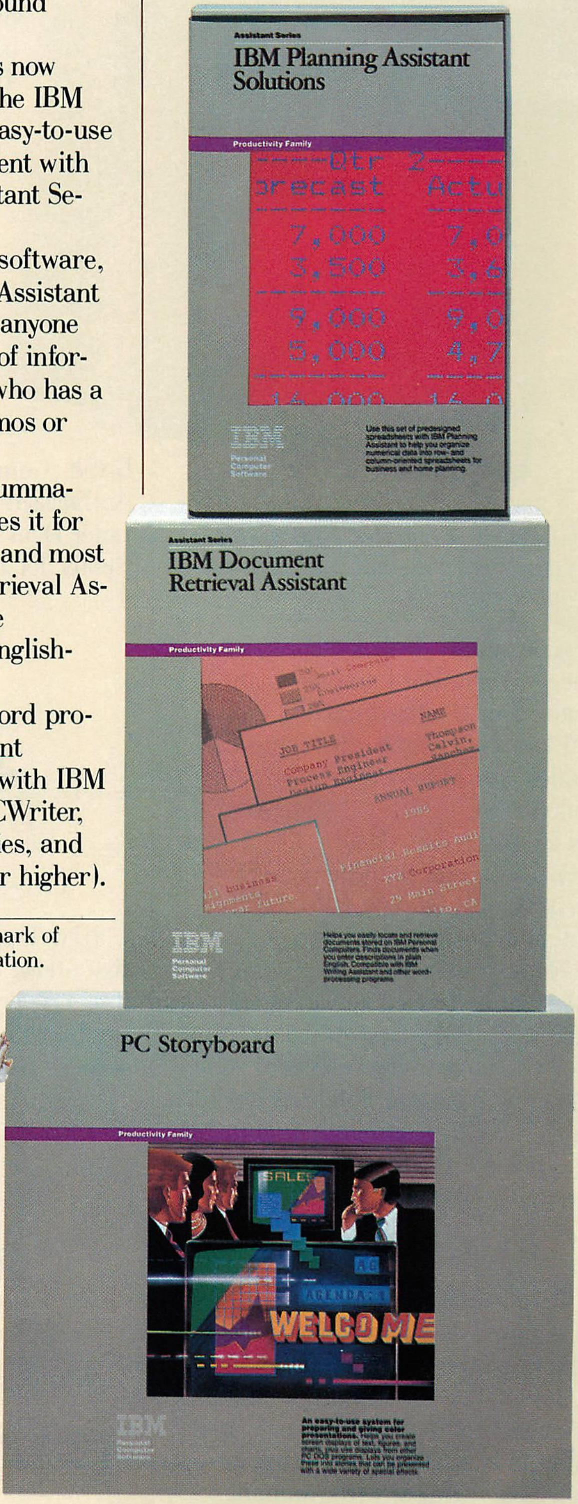
First, it automatically summarizes information and stores it for future reference. Second, and most important, Document Retrieval Assistant allows you to locate information with simple English-language queries.

Among other popular word processing software, Document Retrieval Assistant works with IBM Writing Assistant, IBM PCWriter, the IBM DisplayWrite Series, and WordStar® (version 3.10 or higher).

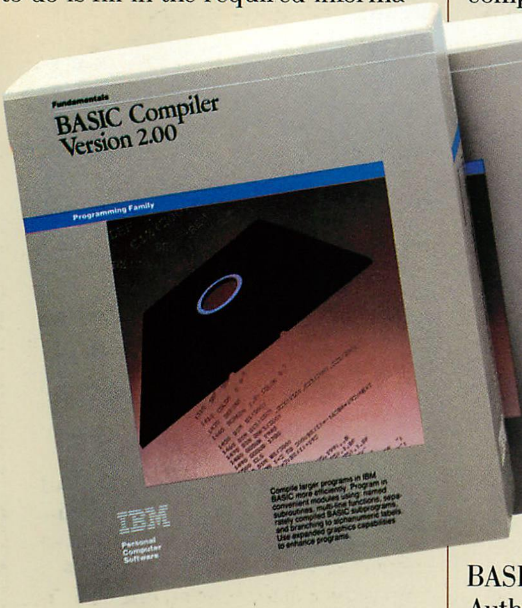
WordStar is a registered trademark of MicroPro International Corporation.

Financial planning.

Life is riddled with seemingly straightforward questions that result in headaches instead of answers. Questions like, "What's my net worth?" and "How much will my monthly payments be?"



IBM Planning Assistant Solutions and IBM Planning Assistant work together to help you find the answers even if you're not familiar with the math involved. All you have to do is fill in the required information on simply designed spreadsheets. Your IBM Personal Computer does the hard part.



IBM BASIC Compiler 2.00 updates an indispensable business programming tool.

Planning Assistant Solutions includes spreadsheets like Loan Calculation and Real Estate Analysis for home use. Others, like Travel Expense Form and Financial Statement Worksheet, can help you sort out your business finances.

Many of the spreadsheets can use information from files you've created with IBM Filing Assistant. You can also include the results of your Planning Assistant Solutions work in IBM Writing Assistant documents and present them graphically with IBM Graphing Assistant.

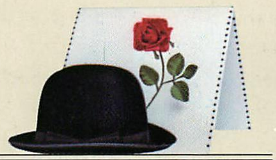
BASIC improvements.

It's good to know that in a business based on technological advances, the basics aren't forgotten.

The IBM BASIC Compiler 2.00 gives exactly that reassurance. It's an improved version of an indispensable business programming tool for the entire Family of IBM Personal Computers.

Among many significant improvements, the most recent version of IBM's BASIC Compiler includes better program control structures, allows you to compile larger programs and supports larger arrays. It also supports the IBM PC Network environment, provides better access to your IBM Disk Operating System (DOS), runs under TopView, and offers expanded graphics capabilities.

And there's a trade-up offer for current licensees that represents a significant savings on the IBM BASIC Compiler 2.00. See your Authorized IBM Personal Computer Dealer or IBM Product Center for details.

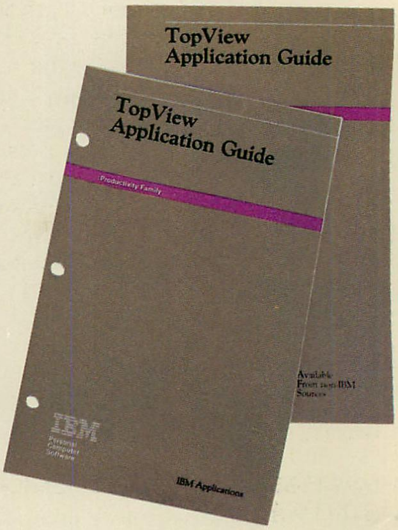


HARDCOPY

Guides to the top.

IBM TopView™ is a program that allows you to run more than one program at a time (multitasking) and to view screens from more than one program on your display (windowing). This, in turn, makes fast work of switching back and forth among programs that you use frequently—word processing, filing and spreadsheet applications, for example.

The TopView Application Guides shown below are now included when you purchase a TopView program. They're useful listings of the many programs—IBM applications and software from non-IBM sources—that can be used with TopView. Both are arranged alphabetically and include special operating considerations and notes whenever necessary.

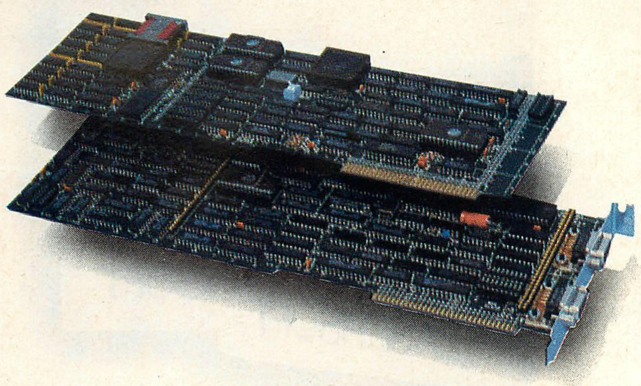


TopView Application Guides now included with the program.

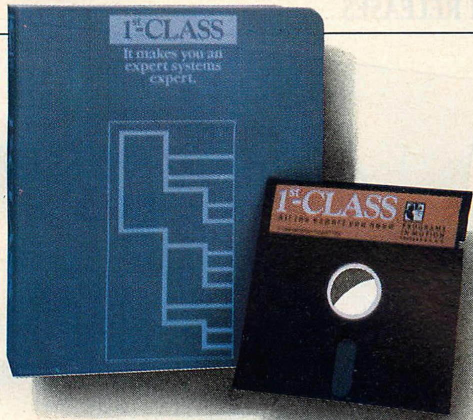
IBM

IBM and Personal Computer AT are registered trademarks of International Business Machines Corporation. Personal Computer XT, PC*jr* and TopView are trademarks of International Business Machines Corporation.

For more information about IBM Personal Computer products discussed in this issue of *Read Only*, see your Authorized IBM Personal Computer Dealer or IBM Product Center. Or call 800-447-4700. In Alaska call 800-447-0890.



Verticom's M-16 and M-256



By Programs in Motion

Verticom, Inc.'s two new high-speed graphics controllers, the **M-16** and the **M-256**, provide the user access to 16 and 256 colors respectively from a palette of 4,096 at a speed of 1 million pixels per second. This drawing capability is generated by a Verticom proprietary two-micron CMOS graphics controller chip set with an on-board Motorola 68000 CPU. M-16 and M-256 are ideally suited for graphic artists, engineers, architects, and other scientific users.

In order to display high-definition, flicker-free, color graphics images with these controllers, Verticom also announced the **CD-1**, a 60-Hz, noninterlaced RGB display with 640-by-480 resolution. Graphics controllers, $2,250 to $2,850; CD-1 $1,025.
Verticom, Inc., 545 Weddell Drive, Sunnyvale, CA 94089; 408/747-1222
CIRCLE 316 ON READER SERVICE CARD

SOFTWARE

Potential REVELATION users can get better acquainted with **Cosmos**'s database and applications environment with **In the Beginning . . . An Introduction to REVELATION**. The program is a functional version of REVELATION; it uses one of that product's common business applications (an accounts receivable system) to explain the concept of data dictionaries and exhibit the capabilities of REVELATION's modules. The demonstration package is designed so users may begin where they wish and proceed at their own pace. $24.95.
Cosmos, Inc., 19530 Pacific Highway S., Suite 102, Seattle, WA 98188; 206/824-9942
CIRCLE 319 ON READER SERVICE CARD

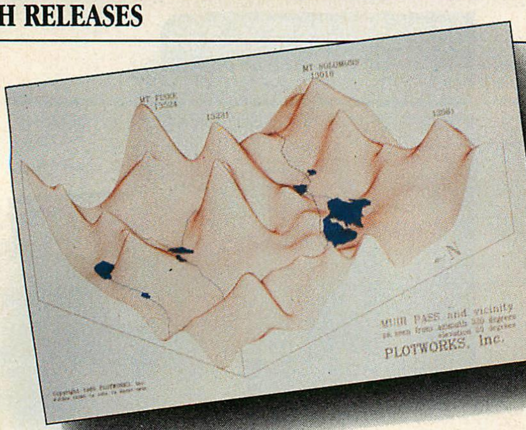
A new test data generator system for UNIX and DOS, called **TDGen Test Data Generator**, has been introduced by **Software Research Associates**. This system simplifies writing files of test data and provides a very flexible method for permitting use of random and sequential generation schemes for test data creation. In addition, TDGen is able to work in multiple stages (including on its own output), which considerably enhances the system's power. The system acts on two files that are created by the user: the file-definition file and the values-definition file. $375.
Software Research Associates, 580 Market Street, San Francisco, CA 94104; 415/957-1441
CIRCLE 320 ON READER SERVICE CARD

SuperSoft, Inc. has released **version 3.0** of its BASIC compiler. The company's latest BASIC is the only PC compiler that meets or exceeds the ANSI X3.60—1978 BASIC standard. It fully supports the 8087 coprocessor and has true IEEE floating-point support. For business applications, in which exact numbers are critical, BCD math can be used to eliminate penny round-off errors. Full double precision is used throughout. SuperSoft BASIC produces fully ROMable code; it lets the user easily link in assembly language or C code, and it produces precise error messages. $300; update fee, $95.
SuperSoft, Inc., 1713 S. Neil Street, P.O. Box 1628, Champaign, IL 61820; 217/359-2112
CIRCLE 321 ON READER SERVICE CARD

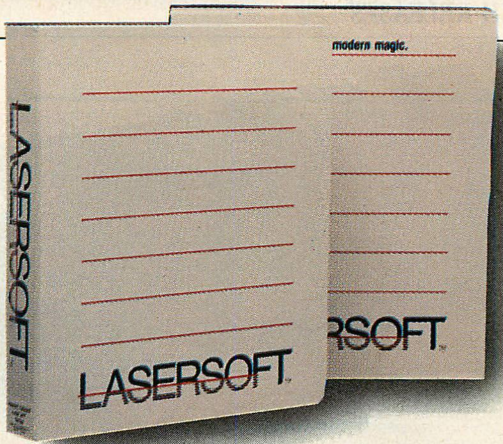
SourceView Software International has introduced **Interactive Cobol Generator** for MS-DOS machines. It runs under Ryan-McFarland COBOL and combines a full-screen editor (for creating RMCOBOL screens) with a complete data dictionary (for maintaining COBOL data definitions in documented form). It can be used with a text editor to provide additional time savings. $149.50.
SourceView Software International, 835 Castro Street, Martinez, CA 94553; 800/443-0100, ext. 440
CIRCLE 324 ON READER SERVICE CARD

Software that permits PC users to build expert systems quickly has been introduced by **Programs in Motion, Inc.** Called **1st-CLASS**, its applications include diagnosing faults, packaging expert advice, correlating financial data, or capturing or calculating logical relationships. 1st-CLASS builds its own rules, based on sample decisions or user-supplied data examples; it can work with other programs to read instruments or control machines to create a closed-loop system that needs no operator. The program has a spreadsheet format. $250; 1st-CLASS Intro, $50.
Programs in Motion, Inc., 10 Sycamore Road, Wayland, MA 01778; 617/879-9650
CIRCLE 326 ON READER SERVICE CARD

Intel Corporation has introduced a programming conversion package that allows users to program high-density plastic EPROMs (erasable programmable read-only memories) much faster. The **iUP-FAST 27K-CON** conversion package incorporates the quick-pulse programming algorithm, which reduces the time required to program a 256-kilobit EPROM from 3 minutes to 30 seconds. The iUP-FAST 27K **Personality Module** works with Intel's line of iUP Universal Programmers and the iPDS development system. It supports Intel's EPROMs, including all the new one-time-programmable (OTP) EPROMs in plastic packages and EPROMs made using Intel's CHMOS technology. An upgraded iUP-FAST 27K module can be used in the stand-alone iUP-201A or under host control with the iUP-200A and iUP201. With a new version of the iPPS PROM programming software, the PC, PC/XT, and PC/AT can be used as programming hosts. iUP-FAST 27K-CON, $150; Personality Module, $845.
Intel Corporation, Literature Department, W251, 3065 Bowers Avenue, Santa Clara, CA 95051; 408/496-8671
CIRCLE 323 ON READER SERVICE CARD



PLOT88 screen



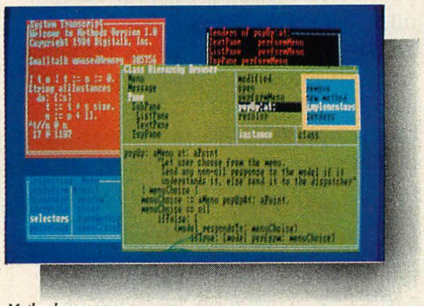
From Business Systems International

Now available from **Programming Concepts, Inc.** is **version 2.0** of its CLIST product. CLIST provides the hardcopy printout and cross-reference that most C compilers neglect; it also is the only software package that calculates and prints metrics and statistics. $129.
Programming Concepts, Inc., 2150 Smithtown Avenue, Ronkonkoma, NY 11779; 516/467-5200
CIRCLE 331 ON READER SERVICE CARD

Network Software Associates, Inc. has announced an addition to its AdaptSNA RJE product series. Called **AdaptSNA RJE/APF** (for remote job entry/automatic processing facility), it allows a remote site PC to function as an intelligent RJE workstation, lets the user automatically determine how to reformat or modify downloaded mainframe data, and allows the communicating PC to drive nonstandard I/O devices. The capabilities of RJE/APF include data format conversion, accommodation of special file formats, index building, restructuring of data into hierarchical or tree files, and predetermined data analysis tasks. $870.
Network Software Associates, Inc., 19491 Sierra Soto, Irvine, CA 92715; 714/768-4013
CIRCLE 325 ON READER SERVICE CARD

PLOT88, from **PLOTWORKS, Inc.**, is a software library that lets the user construct charts, graphs, and contour maps, fully annotated with text drawn in one of seven type styles, at varying height and at an angle. PLOT88 provides pen plotter emulation on IBM PC or Epson printers, IBM display screen, and HP2686A LaserJet printer; it constructs high-resolution drawings on HP plotters. PLOT88 fully supports the 8087 or 80287 coprocessor and employs raster technology to provide maximum through-put. $250.
PLOTWORKS, Inc., P.O. Box 12385, La Jolla, CA 92037-0635; 619/457-5090
CIRCLE 328 ON READER SERVICE CARD

An object-oriented programming environment called **Methods** furnishes PC users with a tool for prototyping solutions to complex problems. Produced by **Digitalk Inc.**, this product uses a mouse, pop-up menus, windows, color, and a virtual machine architecture to develop man-machine interfaces, expert systems, and computer-aided instruction tools. With Methods, PC users now have access to Smalltalk, which is an object-oriented programming language that tries to mirror human ways of thinking and problem solving. $250.
Digitalk Inc., 5200 W. Century Blvd., Los Angeles, CA 90045; 213/645-1082
CIRCLE 339 ON READER SERVICE CARD


Methods screen

Systems software that is fully compatible with the graphics ROM BIOS on the IBM Enhanced Graphics Adapter (EGA) has been introduced by **Phoenix Software Associates Ltd.** The company's **EGA-compatible ROM** will be licensed to IBM-compatible personal computer and add-in board manufacturers who offer EGA-compatible video boards. Phoenix's EGA-compatible ROM BIOS is available for licensing to OEMs on unlimited use fee or royalty basis. The company can provide OEMs with the custom hardware or software engineering services required to install the EGA ROM on an OEM's video board.
Phoenix Software Associates Ltd., 320 Norwood Park S., Norwood, MA 02062; 800/344-7200
CIRCLE 322 ON READER SERVICE CARD

Lasersoft, from **Business Systems International, Inc.** creates complex or simple business forms, presentation graphics including bar graphs, Gannt charts, and organization charts, and any type of diagram composed of vertical and horizontal lines, boxes, shading, and text. Menu-driven, the software integrates computers with electronic laser printers, enhancing the features of both. Other capabilities include merging graphics with text and replicating a computer CRT screen complete with mock screen border and inverse video, half-bright, or other display enhancement. In addition, **Xerox Corporation** has signed a worldwide distribution agreement to market the product. $995.
Business Systems International, Inc., 20942 Osborne Street, Canoga Park, CA 91304; 818/998-7227
CIRCLE 374 ON READER SERVICE CARD
Xerox Corporation, 8200 Brookriver Drive, Dallas, TX 75247; 214/689-6335
CIRCLE 334 ON READER SERVICE CARD

Another program for laser printers, **LaserControl**, has been announced by **Insight Development Corporation.** LaserControl works with virtually any program, computer, and the Hewlett-Packard LaserJet printer to provide Epson, Diablo, Nec, and Qume printer emulation and LaserJet graphics from any Epson graphics program. It can print maximum-width Lotus worksheets and other spreadsheets on 14-inch paper. The LaserControl Menu gives the user complete control of the LaserJet without any complicated escape sequences. LaserControl 100 (software only), $150; LaserControl 200 (software embedded on PC board), $499; LaserControl 300 (stand-alone box for use with any micro-, mini-, or mainframe computer), site licensing available.
Insight Development Corporation, 2005 Vine Street, Suite 4, Berkeley, CA 94709; 415/527-8646
CIRCLE 335 ON READER SERVICE CARD

60-DAY MONEY BACK GUARANTEE

Borland's SuperKey And SideKick Work So Well Together, You'll Hardly Work At All.

AN UNBEATABLE TEAM AT AN UNBEATABLE PRICE! We've teamed the best with the best to make the greatest. The best keyboard enhancer, SuperKey®. The best desktop organizer, SideKick®. The dynamic duo working hand in hand to let you do many different things at once. A way that cuts down the keystrokes, so you're working instead of just typing. A way that wasn't possible until we paired the electronic wizardry of SuperKey with the practical efficiency of SideKick. SuperKey brings the magic. SideKick does the details. The "S-Team" works beautifully together because we designed them that way.

GET SUPERKEY AND SIDEKICK TODAY and you'll have an unbeatable team at an unbeatable price — and a $15 rebate back in the mail.

SIDEKICK INCLUDES: * Calculator * Notepad * Auto dialer & phone directory * ASCII table * Perpetual calendar & datebook * Help window * Full-screen editor with word-wrap, paragraph editing and much much more. (Chances are that once you have SuperKey and SideKick working together for you, you'll never need to use a word-processing program again).

SUPERKEY INCLUDES: *Encryption to keep confidential files confidential * Programmable keys that let you turn a thousand keystrokes into one * Keyboard lock * Automatic turn-off of your screen after a pre-set time so the expensive phosphorus in your monitor's screen isn't etched or ruined * Secret Password protection and more.

IF YOU USE SIDEKICK, YOU NEED SUPERKEY. BECAUSE SUPERKEY AND SIDEKICK CAN MAKE YOUR DAY GO SOMETHING LIKE THIS:

8:00 am. You got to work on time, despite the 44-mph turkey ahead of you in the fast lane. It's spreadsheet time. You hit one key. Lotus 1-2-3 (or whatever) is up and running. (One key, because SuperKey has recorded all the CD\123 <ENTER>123<ENTER> <ENTER>/F <ENTER> R <ENTER> SALES <ENTER> <PgDn> foolishness and your one keystroke played all that back instantly. One keystroke instead of a minuet).

8:03 am. You're into the spreadsheet. Phone rings. You kick in SideKick's Notepad—without leaving your spreadsheet. You talk. You listen to Frank. You make notes that tell you that Frank is upping the numbers from yesterday's order and he needs a new price and delivery date. He wants a meeting. Fast, but when? You have SideKick fire up your Calendar. Time agreed and noted—in SideKick's NotePad. Conversation ends. Your spreadsheet is still there.

8:07 am. You're watching the spreadsheet but you're thinking about the new bid you have to figure out. So you have SideKick's Calculator pulled up on the screen—over a small piece of the spreadsheet—which doesn't go away.

8:08 am. SideKick is coming up with new numbers. SuperKey keeps the spreadsheet on a roll. Satisfied with the numbers, you have SideKick auto-dial Frank's number. Talk. Talk. Hang up.

8:09 am. Spreadsheet about done. You're watching it, but thinking about what Frank just said on the phone. He liked your numbers. He ordered. He said, "That was fast. We won't need that meeting. (SideKick cancels it from your Calendar). And he also said, "How did you get all that done so quickly?" And you said, "I've got a couple of new guys working for me."

IF YOU DON'T USE SIDEKICK YET, YOU GET THEM BOTH AND FOR A LIMITED TIME, A $15.00 CASH-BACK! Because SuperKey and SideKick are so compatible, we let them move in together. Into their own blister-pack. With the $15.00 cash-back coupon and the manuals. Which is what you get for $139.90 instead of the usual $154.90. You need to fill-in the $15.00 cash-back coupon, along with your registration cards and proof of purchase, and mail it back. We'll rush you your $15.00 rebate right away. Rebate offer ends March 31, 1986. (PS: You can still buy SuperKey and SideKick separately. SuperKey **$69.95**, SideKick **$84.95**. Not copy-protected.)

THE CRITIC'S CHOICE

"If you want the ultimate in sophistication, you won't find anything finer on the battlefield right now than Borland's SuperKey and SideKick combo." **LEON A. WORTMAN, InfoWorld**

BORLAND INTERNATIONAL
4585 Scotts Valley Drive, Scotts Valley CA 95066
Phone (408) 438-8400 Telex 172373

NOT COPY-PROTECTED

SuperKey SIDEKICK

SuperKey and SideKick require DOS version 2.0 or later.
Software for your IBM®PC, XT, AT, 3270, PCjr and true compatibles.

Available at better dealers nationwide. Call (800) 556-2283 for the dealer nearest you. To order by Credit Card call (800) 255-8008, CA (800) 742-1133.

Yes, I want the Best!

	Quantity
The SideKick-SuperKey Team **$154.90** $15 cash-back ONLY when purchased together. Both come not copy protected.	
SuperKey **$69.95**	
SideKick **$54.95** (copy protected)	
SideKick **$84.95** (not copy-protected)	

Subtotal
(CA res. add 6% tax)
Amount Enclosed:
Payment VISA MC BankDraft Check
Credit Card Exp. Date /
Card #

$15.00 Cash Back

SuperKey only—Outside USA and Canada: Due to US Government restrictions on export of technology, purchase of SuperKey (without DES) can only be made through authorized distributors.

SideKick only: PCjr requires the not copy-protected version.

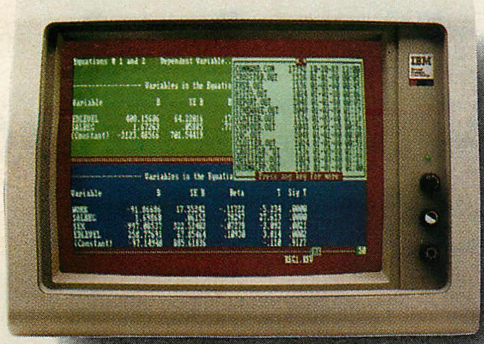
$15 cash-back rebate upon receipt of signed license agreement, cash-back coupon and proof of purchase. Your $15 cash-back rebate includes tax when applicable. These prices include shipping to all U.S. cities. All foreign orders add $10 per product ordered.

Name:
Shipping Address:
City:
State: Zip:
Telephone:

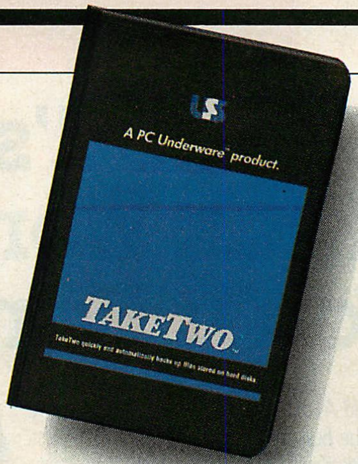
COD's and Purchase Orders WILL NOT be accepted by Borland. California residents: add 6% sales tax. Outside USA: add $10 and make payment by bank draft, payable in US dollars drawn on a US bank.

Y-6

CIRCLE NO. 251 ON READER SERVICE CARD



SPSS/PC+ screen

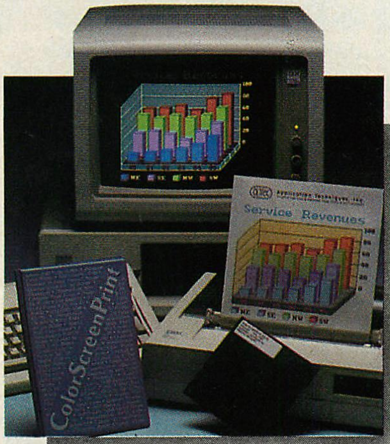


By United Software

Software Intelligence Laboratory, Inc. has introduced **WIZDOM PX**, the newest member of its expert system family. Functionally similar, WIZDOM PX has a new feature called the program function memron (PFM) which enables it to take full advantage of both conventional software program and expert system technology. PFM can be used to communicate with other PCs or hosts; alternatively it can be used for realtime functions with many engineering applications. A PFM is a memron existing within the semantic network inside WIZDOM's knowledge base and can represent a piece of conventional software. License fee, $2,050.
Software Intelligence Laboratory, Inc., 50 Broad Street, 10th floor, New York, NY 10004, 212/747-9066
CIRCLE 336 ON READER SERVICE CARD

A new database management system from **U.S. Video** overlays database information onto a video image and performs all database functions. **Video Database Management** (VDM) couples the data/text storage and retrieval power of current database managers with the random access display capabilities of a laser disc player. The user can access either still images or complete movie segments from within the database program. In addition, the VDM can be linked to a video camera, a VCR, or a television signal. The package includes a U.S. Video Raster Master board and a driver program. $980.
U.S. Video, 1730 Pennsylvania Avenue NW, Washington, DC 20006; 202/783-7830
CIRCLE 337 ON READER SERVICE CARD

Application Techniques, Inc. has announced **ColorScreenPrint 2.0**, a new release supporting 50 printer models that lets PC/XT and PC/AT users create a color printout of their screen at any time, simply by pressing the Shift and PrtSc keys. The program is compatible with most color graphics cards and applications software. Once loaded into the computer's memory, it remains in the background, ready to work with other programs such as Lotus 1-2-3, Framework, and PC Paint. $79.95.
Application Techniques, Inc., 10 Lomar Park Drive, Pepperell, MA 01463; 617/433-5201
CIRCLE 332 ON READER SERVICE CARD



By Application Techniques

A new automatic hard-disk backup system, **TAKETWO**, and a memory-resident utility, **PrivateEye**, have been announced by **United Software Security, Inc.** TAKETWO requires no additional drives or other hardware. After initial installation, the user simply types TT once a day and TAKETWO automatically backs up in minutes all files that have been added or changed during the day. PrivateEye clears the screen with a simple keystroke, assuring privacy of sensitive or personal information. The touch of a key restores the screen just as the user left it. TAKETWO offers annual site or corporate licensing, $1,400; PrivateEye license fee, $300.
United Software Security, Inc., 6867 Elm Street, Suite 100, McLean, VA 22101; 800/892-0007
CIRCLE 327 ON READER SERVICE CARD

SPSS Inc. has released a new version of its SPSS/PC software for the PC/XT and PC/AT. **SPSS/PC+** offers enhanced features to JOIN, ADD, MERGE, and SORT files; a MANAGER utility that allows more user flexibility in managing the workspace environment; and a REVIEW on-line editor that uses a split screen to edit listing files. **SPSS/PC+ TABLES** displays and summarizes data for using tabular form and **SPSS/PC+ ADVANCED STATISTICS** performs multivariate procedures. SPSS/PC+, $795; SPSS/PC+ TABLES and SPSS/PC+ ADVANCED STATISTICS, $295 each.
SPSS Inc., 444 N. Michigan Avenue, Chicago, IL 60611; 312/329-2400
CIRCLE 338 ON READER SERVICE CARD

Disk Optimizer from **SoftLogic Solutions, Inc.** is a performance enhancer for disk storage. It collects the pieces of a file and physically organizes them on the drive so similar information is stored together, according to the subdirectory. Free space is made contiguous so new data are written quickly. Included is the Disk Analysis program, which helps the user determine the drive's optimization percentage. $49.95.
SoftLogic Solutions, Inc., 530 Chestnut Street, Manchester, NH 03101; 800/272-9900
CIRCLE 330 ON READER SERVICE CARD

LeBlond Software, Inc. and **Lotus Development Corporation** have announced **Basic Concerto**, LeBlond's software product that allows developers to design Symphony add-ins using IBM BASIC. With Concerto, users create Symphony add-ins without having to know assembly language. $99.
LeBlond Software, Inc., 2421 Willowbrook Parkway, Indianapolis, IN 46205; 800/222-4711
CIRCLE 373 ON READER SERVICE CARD
Lotus Development Corporation, 55 Cambridge Parkway, Cambridge, MA 02142; 617/577-8500
CIRCLE 329 ON READER SERVICE CARD

AZTEC

Manx Aztec C86 is the best C for MS-DOS and you can prove it yourself!

"A compiler that has many strengths ... quite valuable for serious work"
Computer Language review, February 1985

Manx Aztec C86 - The C For MS-DOS

Manx Aztec C86 is clearly the best C software development system for MS-DOS. Aztec C86 is the only C compiler for MS-DOS that provides the level of performance, features, documentation, and support required for serious, professional software development. You can prove it yourself. All you have to do is order Aztec C86 from Manx, evaluate it, and, if you like it, keep it. If you don't like it, send it back within 30 days and we'll cancel your order.

If you keep your Manx Aztec C86, as 99% do, you'll be in with the best company.

Manx Aztec C86 Features:

Optimized C compiler: Unsurpassed for code quality and speed. Optionally generates 80186 and 80286 code. Full K & R.

Symbolic Debugger: Execution trace, break points, display data in floating point, integer, character, or hex format. Evaluate expressions. Detect illegal memory stores, modify memory/registers, disassemble code.

Manx AS86 Macro Assembler: Supports macroes, 8086, 80186, and 80286 instructions in Intel format. Fast execution.

LN86 Overlay Linker: Links small, large, and mixed memory model routines, supports overlays, and options for producing ROM based code.

Librarian: Build and modify personal or system run time libraries.

8087/80287 Sensing Library: One library simulates floating point, another assumes the presence of an 8086 or 80287 math chip, the third senses the existence of a math chip, and if it finds one it uses it.

Profiler: Provides a run time analysis of your code to pinpoint code segments to optimize.

UNIX Library: Compatible with UNIX C. Fast I/O. Terminal I/O can be buffered or unbuffered.

DOS Library: Time and date functions, program forking (exec), program chaining, directory commands, I/O port support, sysint support, BIOS functions, and BDOS functions.

Screen & Graphics Library: Screen and cursor functions. Fast routines for drawing lines, circles, elipses, points, and setting colors.

CP/M-86 Library (-c): Produce programs for CP/M-86.

Large Memory Model: Manx Aztec C86 supports programs and data of any size. Global data has a max size of 64k.

Intel Object Option: Interface to software that requires Intel object format, such as PLINK86.

Z (vi) Source Editor (-c): Fast, powerful editor, Macro capabilities, undo, ctags, buffers for commands and data, and all the bells and whistles that make vi fanatics fanatical.

ROM Support Package (-c): Startup routine, linker options for separate placement of code and data, special utilities like the Intel HEX Utility, documentation, and library source.

Library Source Code (-c): UNIX, screen, graphics, and math function libraries.

Mixed Memory Models (-c): Mix large code and small data, small code and large data, or mix within type.

UniTools (-c): The UNIX utilities make, diff, and grep.

One year of updates (-c): As new versions are released, updates are automatically sent.

Technical Support: Manx has a full time staff to provide support via telephone & bulletin board.

Items marked -c are special features of the Aztec C86-c system.

Manx Aztec C86 is available in four configurations: Manx Aztec C86-c, Manx Aztec C86-d, Manx Aztec C86-p, and Manx Aztec C86-a. The -p and -a systems are not intended for commercial work and do not incorporate the same compilers as the -c and -d systems. All systems are upgradable.

Aztec C86-c Commercial System	**$499**
Aztec C86-d Developer's System	**$299**
Aztec C86-p Personal System	**$199**
Aztec C86-a Apprentice System	**$ 49**

Manx Cross Development Systems

Manx Aztec C compilers are available as native or as cross development systems for PC-DOS, MS-DOS, Macintosh, CP/M-86, CP/M-80, TRSDOS, Apple II, and Commodore 64/128.

Cross development involves two computer systems: the development system (HOST) and the execution system (TARGET). This method is useful when the TARGET machine is slower or more limited than the HOST.

HOSTS: VAX UNIX ($3000), PDP-11 UNIX ($2000), MS-DOS ($750), CP/M ($750), Macintosh ($750), CP/M-68k ($750), XENIX ($750).

TARGETS: MS-DOS, CP/M-86, Macintosh, CP/M-68k, CP/M-80, TRS-80 3 & 4, Apple II, Commodore C64, 8086/80x85 ROM, 68xxx ROM, 8080/8085/Z80 ROM, 65xx ROM.

Additional TARGETS are $300 to $500 (non VAX) or $1000 (VAX). Call for information, on cross development to the 68000, 65816, Amiga, C128, CP/M-68K, VRTX, and others.

How To Become a Manx Aztec C User

Call 1-800-221-0440 or 1-800-832-9273 (800-TEC WARE). In NJ or outside the USA call 201-530-7997. Orders can also be telexed to 4995812.

Payment can be by check, COD, American Express, VISA, Master Card, or Net 30 to qualified customers.

Orders can also be mailed to Manx Software Systems, Box 55, Shrewsbury, NJ 07701.

For More Information: call 1-800-221-0440, or 201-530-7997, or write to Manx Software Systems.

Manx maintains a large professional staff to service and support Manx users. You will get fast delivery and great service dealing directly with Manx.

Support Software for Manx Aztec C86

C-tree $395: B-tree database system. Easy to use. Available for Aztec C for MS-DOS, Macintosh, CP/M-86, CP/M-80, and others. Includes source.

PHACT $250: Powerful database system. Available for Manx Aztec C compilers for MS-DOS, CP/M-86, CP/M-80, and Macintosh.

PANEL $295: Create screens via simple, powerful editing commands. Select colors, edit fields. Directly input data to a multi-keyed file utility included with the system.

SunScreen $99: Create and modify formatted screens easily. Validate fields, select colors, create screens for both the color and monochrome cards. With library source SunScreen is $199.

WindScreen $149: Combines SunScreen with a powerful window utility.

Windows for C $195: Versatile window utility that supports IBM PC compatible and some non-compatible environments.

AMBER Windows $99: Powerful, low priced window package.

HALO $250: The ultimate C graphics package. It supports viewports, shapes, and multiple graphics cards. A less expensive version is available for just the PC mono and color cards.

FirsTime $295: Syntax checking while you edit greatly shortens compile time.

Pre-C $395: Powerful Lint-like utility locates structural and usage errors. Easily checks multiple files for bad parameter declarations and other interface errors. Lint users will find the user interface a dream come true.

PC-LINT $98: Lint-like utility that supports large memory models, has clear error messages, and executes quickly, has lots of options and features that you wouldn't expect at this low price.

Greenleaf Functions $185: Source for over 200 C and assembler functions. They are great, they work, they are used extensively, and are economically priced. Clear documentation and easy to use interface round out an impressive package.

C Utility Library $185: C and assembler source for screens, windows, color graphics, asynch communications, and more. The color graphics and speed of this package are impressive.

Plink-86 $395: MS-DOS linkage editor for producing and maintaining overlayed programs. It works with Aztec C86 in Intel object format mode.

30 Day Guaranty:

Any Manx Aztec development system can be returned within 30 days for a refund if it fails to meet your needs. Restrictions are that the original purchase must be directly from Manx, shipped within the USA, and the package must be in new condition. Returned items must be received by Manx within 30 days. A restocking fee may be required.

Discounts:

There are special discounts available to professors, students, and consultants. A discount is also available on a "trade in" basis for users of competing C systems.

Manx Aztec C Distribution:

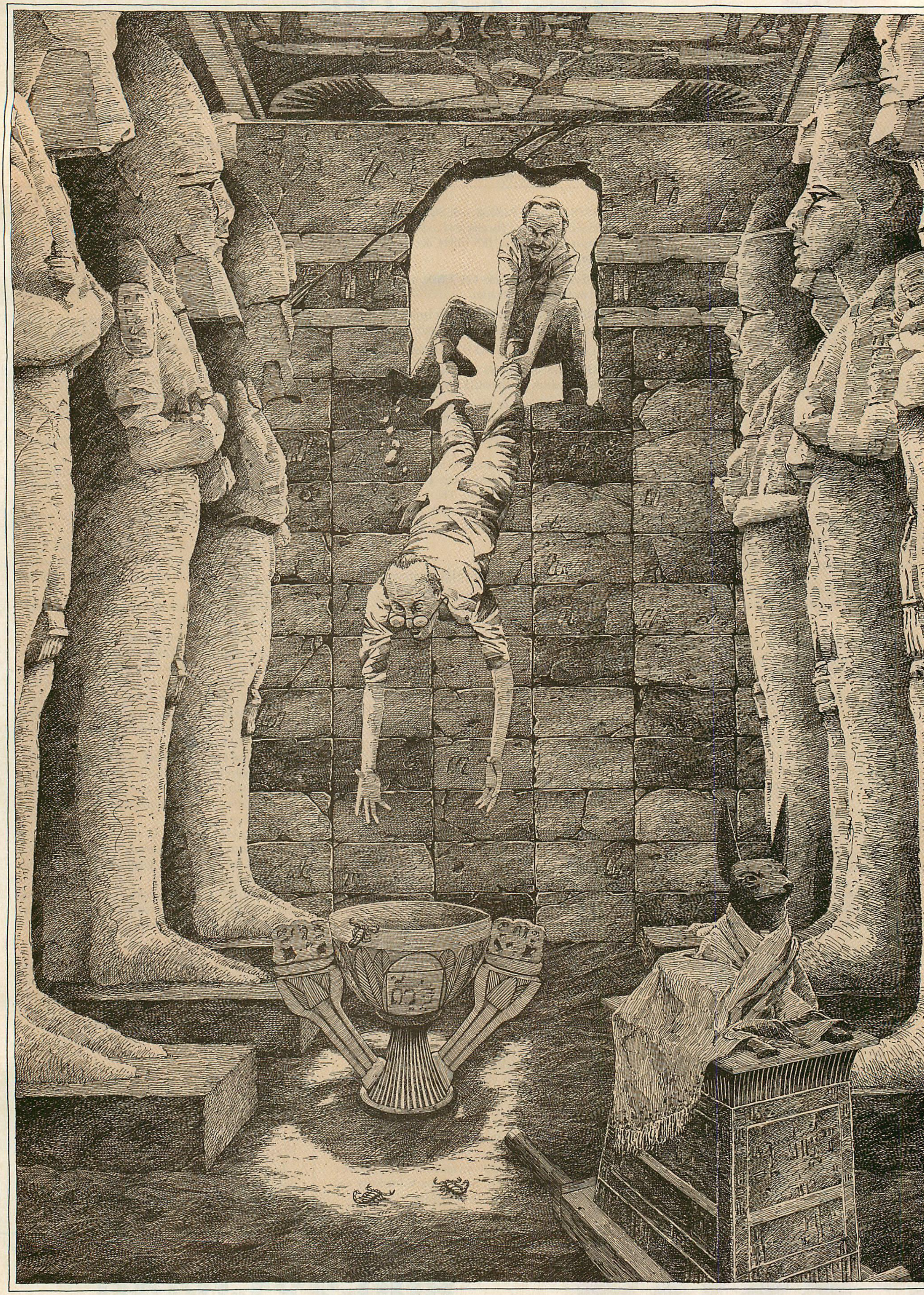
In the USA, Manx Software Systems is the exclusive distributor of Aztec C. Telephone or mail order sales other than through Manx are unauthorized.

MANX®

To order or for information call:

800-221-0440

CIRCLE NO. 208 ON READER SERVICE CARD



COMMON SENSE IN DATABASE MANAGEMENT
NUMBER FOUR IN A SERIES.

IT DOESN'T MATTER HOW GOOD IT IS IF YOU CAN'T GET YOUR HANDS ON IT.

Buried deep in your company, there's an incredible source of wealth. People have spent thousands of hours creating it, one painstaking step at a time.

Then they store it in places where it'll never see the light of day. As you know, we're talking about data.

Because in most companies, the information you need to build a database is locked up in the corporate mainframes, minis and micros. Stored on hundreds of tapes and floppy disks. In a dozen different programs. And with most database software for PCs, the only way to put all this information together is to key it in stroke by stroke. Or write a different program for every data transfer. Unless you've got R:base™ 5000.

ACCESSIBLE DATA

Using R:base 5000, it's possible to transfer data from any of the following popular programs and files. Instantly. Using dBASE III,™ it's not nearly as easy.

Automatic Data Transfer From:	R:base 5000	dBASE III
Lotus 1-2-3®	Yes	No
pfs:file®	Yes	No
dBASE II®	Yes	Yes
Multiplan® (Sylk)	Yes	No
Visicalc® (Dif)	Yes	No
ASCII (delimited and fixed files)	Yes	No

AUTOMATIC DATA TRANSFER. ONLY WITH FILEGATEWAY.™

With some DBMS software, you have to spend 90% of your time copying data that's already been keyed in somewhere else. But R:base 5000 has a FileGateway™ feature. Which lets you transfer information from all the programs listed in this chart. Automatically, without programming. Without even defining a file structure. All you do is follow a simple menu.

As a result, you can build a database with thousands of records in about the same time it takes to copy a disk.

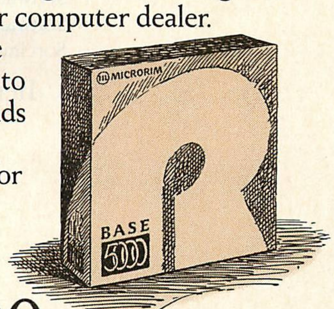
And once you've got all that data in R:base 5000, you can really put it to good use. Because it lets you develop applications ten times faster, runs 60% to 150% faster, and gives you a much more sophisticated report writer than dBASE III.

SEE FOR YOURSELF: 1-800-547-4000.

If you believe in common sense as much as we do, you'll want to try it out and see for yourself. So for only $9.95 (plus shipping) we'll send you a mini-version of the product that lets you build real applications.

Just call 1-800-547-4000 and ask for Dept. 853. From Oregon, or outside the U S., call 1-503-684-3000, Dept. 853. We'll send your copy right out. If you'd like to see R:base 5000 today, head straight for a leading software store or computer dealer.

And see what it's like to get your hands on a little satisfaction for a change.



R:BASE 5000
FROM MICRORIM

IT ALL COMES DOWN TO COMMON SENSE.

RUNTIME VERSION NOW AVAILABLE.

The following trademarks are acknowledged: Lotus 1-2-3 and Visicalc—Lotus Development Corporation; dBASE II and dBASE III—Ashton-Tate; Multiplan—Microsoft Corporation; pfs:file—Software Publishing Company.

If you want to run software with graphics on your monochrome monitor, we have some bad news.

As we're sure you've been told, the only way to run software with graphics on a monochrome monitor is to buy a graphics card.

For $499, the Hercules Graphics Card runs these best-selling programs:

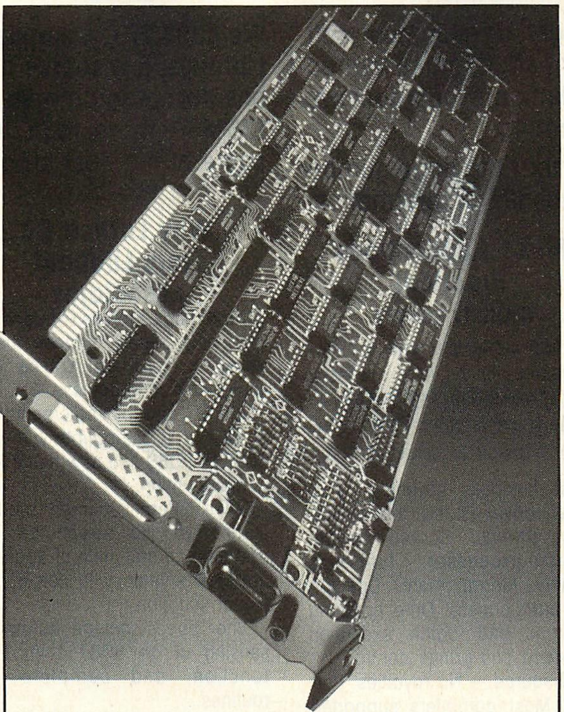
Ashton-Tate, *Framework*
BPS, *Overhead Express*
Lotus Development, *1-2-3, Symphony*
MicroSoft, *MicroSoft Flight Simulator, MicroSoft Word, MicroSoft Chart*
Software Products Int., *Open Access*
Software Publishing, *PFS: Graph*
Sorcim, *Supercalc 3*

In monochrome only.

And some good news.

For $395, the Paradise Modular Graphics Card runs all these programs. In monochrome. And in color.

Arrays/Continental Software, *Ultra File*
Ashton-Tate, *Framework*
BPS, *Overhead Express*
Brightbill-Robert, *Graphix Partner*
Chang Labs, *GraphPlan*
Dow Jones & Co., *Dow Jones Market Analysis*
Lotus Development, *1-2-3, Symphony*
MicroPro, *Chartstar, Planstar*
MDBS, *Knowledge Manager*
MicroSoft, *Basic Compiler, Basic Interpreter, Chart, Flight Simulator, Project, Word*
PC Software of San Diego, *Executive Picture Show*
Prentice-Hall, *Execuvision*
Schuchardt Software Systems, *Intecalc, Intemate, Intepert, Inteplan, Inteword*
Softrend, *Aura*
Software Products Int., *Open Access*
Software Publishing, *PFS:Graph*
Sorcim, *SuperCalc 3*
Summa Software, *Winning On Wall Street: Trader's Forecast, Winning On Wall Street: Trader's Data Manager*
Advanced Ideas, *The Game Show, Master Match, Tic Tac Show*
CBS Software, *Big Bird's Special Delivery, Dinosaur Dig, Ernie's Magic Shapes*
Davidson & Associates, *Math Blaster!, Word Attack!*
Designware, *The Grammar Examiner, Math Maze, Language Arts, Spellicopter, States & Traits, Trap-a-zoid*
Developmental Learning Materials, *Alien Addition, Alligator Mix, Demolition, Division, Dragon Mix, Meteor Multiplication, Minus Mission*
Eduware, *Algebra 1, Algebra 2, Algebra 3, Algebra 4, Algebra 5*
Individual Software, *Professor Pixel, The Instructor, The Typing Instructor*
Knoware, *Knoware*
Scarborough Systems, *Mastertype, Songwriter*
Learning Co., *Addition Magician, Magic Spells, Moptown Parade, Number Stomper, Reader Rabbit*
Scholastic, *Turtle Tracks*
Sierra On-Line, *Dragon's Keep, Troll's Take*
Simon & Schuster, *Typing Tutor III*
Spinnaker Software, *Alphabet Zoo, Delta Drawing, Fraction Fever, Hey Diddle Diddle, Kids on Keys, Kindercomp, Rhymes & Riddles, Story Machine*



The Paradise Modular Graphics Card runs all the popular programs Hercules does, plus 150 more.

Springboard, *Early Games for Young Children, Make a Match, Piece of Cake Match, Quizagon*
Thoroughbred Software, *Exploring the Amazing Food Factory, The Fascinating Story of Cell Growth, How Plants Grow, Migrating Molecules, Mastering Units of Measurement, Photosynthesis*
Unicorn Software, *Funbunch, Ships Ahoy, Ten Little Robots*
Digital Research, *DR Logo*
Energtonics, *Energraphics*
Fox & Geller, *dGraph, OZ*
Graphic Communication, *Graphwriter BASIC, Graphwriter Combination, Graphwriter Extension*
Harvard Associates, *P.C. Logo*
Innovative Software, *Fast Graphs*
Mouse Systems, *PC Paint*
PC Software of San Diego, *PC Crayon*
Peachtree Software, *Business Graphics System*
Arktronics, *Jane*
Eagle Software Publishing, *Personal Financier*
Monogram, *Dollars and $ense*
Penguin Software, *Graphics Magician*
Sierra On-Line, *Homeword*
Adventure Enterprises, *Sea Dragon*
Atarisoft, *Centipede, Defender, Dig Dug, Donkey Kong, Pac Man, Robotron, Stargate*
Avalon Hill Game Company, *Andromeda Conquest, Computer Football Strategy, Computer Stocks & Bonds, V.C., Voyager*

Broderbund Software, *Serpentine*
CBS Software, *Match-Wits, Mystery Master: Murder by the Dozen*
Hayden Software, *Sargon III*
Innovative Design Software, *Pool 1.5*
Intelligent Statements, *Asylum*
Microlab, *Crisis Mountain, Death in the Caribbean, Dino Eggs, High Rise, Miner 2049er*
Muse Software, *Castle Wolfestein*
Odesta, *Backgammon, Checkers, Chess, Odin*
Origin Systems, *Ultima III*
Orion Software, *J-Bird*
PC Software of San Diego, *Championship Blackjack*
Penguin Software, *The Quest*
Priority Software, *Forbidden Quest*
Scarborough Systems, *Buck Rogers, Congo Bongo, Star Trek*
Sentinent Software, *Cyborg*
Sierra On-Line, *BC's Quest of Tires, Championship Boxing, CrossFire, Dark Crystal, Frogger, King's Quest, Oil Well, Ultima II, Ulysses and the Golden Fleece*
Sir Tech, *Wizardry*
Sirius Software, *Buzzard Bait*
Spectrum Holobyte, *Gato*
Spinnaker Software, *Snooper Troops #1, Snooper #2*
Sublogic, *"Night Mission" Pinball*

It's true, Hercules only runs 10 of the 161 programs with graphics for the IBM PC carried by SOFTSEL®, the largest distributor of micro computer products.

Since the Paradise Modular Graphics card is 100% compatible with the IBM color graphics standard, it'll run virtually every program written for the PC. In monochrome. And in color. Now and in the future.

And we give you a $50 trade-in allowance on your old Hercules or IBM card.

So see your dealer or call us. And get some good news for a change.

PARADISE
SYSTEMS, INC

CIRCLE NO. 230 ON READER SERVICE CARD

Paradise Systems Inc., 217 East Grand Ave., South San Francisco, CA 94080
(800) 822-2020 Ext 210 (CA) or (800) 527-7977 Ext 210 (Outside CA)

PAUL PIERCE

The Dashed Cursor

A bug in the BIOS can cause some programs to turn the DOS underline cursor into a dash.

Strangely enough, programs run on a monochrome display sometimes return the cursor to the screen as a dash instead of an underline. Users may wonder why the programmer deliberately changed the cursor in such an odd way, but in all probability the change was not deliberate; the program, in fact, undoubtedly did its best to keep the cursor from changing. Before the program was run, it asked the BIOS what the cursor looked like, then, when the program finished, it restored a cursor with that same appearance. The problem is that the BIOS video call read-cursor-position (interrupt 10H, AH=3), which is supposed to return the current cursor mode in CX, does not always tell the truth.

The confusion starts when the BIOS sets the video mode for the display. The cursor mode also is initialized at this time and stored in two places. First, it is set in the hardware of the video adapter so that the cursor will be displayed properly on the screen. Second, a copy of the cursor mode is stored in RAM. The BIOS video call set-mode (interrupt 10H, AH=0), which is used when the system is started as well as by the DOS mode command, correctly sets the hardware cursor parameters for the color or monochrome display adapter; however, it invariably sets the cursor mode in RAM as if the color adapter were being used. When executed, the read-cursor-position call returns this RAM copy of the cursor mode, which is correct, of course, if the user has a color adapter but incorrect if he is using a monochrome adapter.

For the color adapter the cursor occupies scan lines 6 and 7 of a character position, while the monochrome adapter uses scan lines 11 and 12. Thus, trying to use a cursor on a monochrome adapter with parameters set for a color adapter produces a cursor that is too high and looks like a blinking dash instead of a blinking underline.

The BIOS video call set-cursor-type (interrupt 10H, AH=1) does not have the bug found in the set-mode call. When a program calls set-cursor-type, both the adapter parameters and the RAM copy are set correctly. The IBM Personal Editor is an example of a program that always uses set-cursor-type. This means that, with the Personal Editor, the blinking dash problem is intermittent. If a program is run immediately after the system has been rebooted or immediately after the mode command has been executed, the blinking dash may appear because the copy of the cursor mode that is stored in RAM is incorrect. If, however, the same program is run after the Personal Editor has been used, the blinking dash will not appear because the cursor mode now stored in RAM has been corrected.

The listing below shows the FIXCURS.ASM program, which detects and corrects an incorrect RAM cursor mode. The program first checks to see if the system is using a monochrome adapter; it then checks to see if the cursor mode returned by read-cursor-position is that of a color adapter. If these two conditions hold, FIXCURS uses set-cursor-type to correct the problem. This program can be used alone in an AUTOEXEC.BAT file or the code can be included as part of a larger program that would otherwise restore a blinking dash to the screen after execution.

Paul Pierce is an independent software consultant specializing in IBM PCs and Intel microprocessors. His company, Data Exchange, is based in the "Silicon Forest" near Portland, Oregon.

LISTING: FIXCURS.ASM

```
CODE    SEGMENT PUBLIC 'CODE'
        ASSUME  CS:CODE, DS:CODE, ES:NOTHING

;
;       This program is set up to be made into a COM file
;

        ORG     100H

;
;       First check for the monochrome adapter.
;

START:  INT     11H             ; SET AX = EQUIPMENT FLAGS
        AND     AL, 30H         ; MASK OFF ALL BUT VIDEO BITS
        CMP     AL, 30H         ; TEST FOR MONOCHROME ADAPTER
        JNE     EXIT            ; JUMP IF NOT MONOCHROME

;
;       Now check for incorrect cursor mode returned from the BIOS
;

        MOV     AH, 3           ; CALL BIOS TO GET CURSOR TYPE
        INT     10H             ;
        CMP     CX, 0607H       ; CHECK FOR INVALID (COLOR) TYPE
        JNE     EXIT            ; JUMP IF NOT A BAD VALUE

;
;       At this point we know that the monochrome adapter is
;       in use and that the BIOS cursor mode is incorrect.
;
;       Call the BIOS to set the cursor type correctly.
;

        MOV     CX, 0B0CH       ; USE CORRECT CURSOR TYPE
        MOV     AH, 1           ; CALL BIOS TO SET CURSOR TYPE
        INT     10H             ;

EXIT:   MOV     AH, 0           ; EXIT BACK TO DOS
        INT     21H             ;

CODE    ENDS

        END     START
```

PROGRAMMER DEVELOPMENT TOOLS

We Accept
Corporate Accounts & P.O.'s
from qualified companies, educational institutions and government agencies
at NO EXTRA CHARGE.

And to serve our
West Coast Customers
Even Better, We're Now Staying Open Until 5PM Pacific Time
(8 PM Eastern).

APL AND UTILITIES

	LIST	OURS
APL*PLUS/PC System by STSC	595	**449**
APL*PLUS/PC Tools Vol 1 by STSC	295	**239**
APL*PLUS/PC Tools Vol 2 by STSC	95	**75**
APL2C by Lauer Software ..... *Interfaces APL to C*	150	**139**
Financial/Statistical Library by STSC	275	**219**
Pocket APL by STSC ..... *New & Better*	95	**89**
STATGRAPHICS by STSC	695	**539**

ARTIFICIAL INTELLIGENCE

Experteach by Intelliware .... *Complete System*	475	**399**
EXSYS *Expert System Development Software*	395	**339**
Golden Common LISP by Gold Hill	495	**Call**
Insight I by Level Five Research .... *AI Primer*	95	**79**
Insight II by Level Five Research	485	**399**
Methods *Prototyping Utility* by Digitalk	250	**219**
muLISP by Microsoft	250	**189**
Prolog-86 by Solution Systems	125	**Call**

ASSEMBLERS AND DEBUGGERS

8088 Assembler *w/Z-80 Translator* by 2500 AD	100	**89**
Advanced Trace-86 by Morgan Computing	175	**149**
Codesmith-86 *Debugger* by Visual Age	145	**129**
Microsoft Macro Assembler *w/utilities*	150	**109**
Pasm86 *High Perf Assembler* by Phoenix	295	**219**
Periscope Debugger by Data Base Decisions	295	**269**
Periscope II by Data Base Decisions .... *New*	145	**129**
Pfinish *Performance Analyzer* by Phoenix	395	**289**
Pfix-86 Plus *Symbolic Debugger* by Phoenix	395	**289**
Profiler by DWB Associates .... *New version*	125	**109**
Turbo EDITASM *Fast Assembler* by Speedware	99	**89**
Visible Computer: 8088 by Software Masters *New*	70	**59**

BASIC LANGUAGE

BetterBASIC by Summit Software	200	**169**
8087 Math Support	99	**89**
Btrieve Interface .... *Used with Btrieve*	99	**89**
Run-time Module	250	**239**
Microsoft QuickBASIC Compiler .... *New*	99	**79**
Professional BASIC by Morgan Computing	99	**79**
8087 Math Support	50	**47**
True Basic from Addison-Wesley	150	**119**
Run-time Module	500	**459**

BLAISE PRODUCTS

Asynch Manager *for C or Pascal*	175	**139**
C Tools	125	**109**
C Tools 2	100	**89**
Exec Linker *Program Chainer*	95	**84**
Pascal Tools	125	**109**
Pascal Tools 2	100	**89**
Turbo ASYNCH *for Turbo Pascal*	100	**89**
Turbo POWER TOOLS *for Turbo Pascal*	100	**89**
View Manager *for C or Pascal*	275	**219**
View Manager *with Source*	425	**329**

C COMPILERS

	LIST	OURS
C-86 Compiler by Computer Innovations	395	**299**
DeSmet C Compiler *w/Source Debugger . New version*	159	**145**
Lattice C Compiler from Lattice	500	**349**
Lattice C from Lifeboat	500	**299**
Let's C Compiler by Mark Williams .... *New*	75	**69**
Let's C with csd *Source Level Debugger* .... *New*	150	**129**
MWC-86 version 3 by Mark Williams	495	**379**
Microsoft C Compiler	395	**259**
Wizard C by Wizard Systems .... *Includes lint*	450	**369**

MICROSOFT C Compiler

This advanced, professional C compiler system features small, medium and large memory models, a linker, a librarian, extensive documentation, wide support of C add-on libraries, Xenix compatibility and much more.

List Price $395 Our price **$259**

C INTERPRETERS

C-terp by Gimpel Software	300	**249**
Instant C by Rational Systems	500	**379**
Introducing C by Computer Innovations	125	**109**
Run/C from Lifeboat	150	**109**
Run/C Professional from Lifeboat	Call	**Call**

C UTILITIES

Call us for availability of specific compiler interfaces.

APT Toolkit by Shaw American Technology	395	**339**
Asynch Manager by Blaise Computing	175	**139**
Basic_C Library by C Source	175	**139**
Btrieve by SoftCraft	250	**199**
C Cross Reference Generator By Lattice	50	**45**
C-Food Smorgasbord by Lattice	150	**119**
C Power Packs by Software Horizons	Call	**Call**
C-Sprite *Program Debugger* by Lattice	175	**149**
C to dBase by Computer Innovations	150	**139**
C Tools by Blaise Computing	125	**109**
C Tools 2 by Blaise Computing	100	**89**
c-tree by FairCom	395	**329**
C Utility Library by Essential Software	185	**139**
CI Probe *Source Level Debugger* .... *New*	225	**199**
CI ROM Pack *for C-86*	195	**149**
Curses *Screen Manager* by Lattice	125	**109**
Curses *with source code*	250	**219**
dBC *dBase File Manager for C* by Lattice	250	**219**
dBC *with source code*	500	**429**
db_VISTA *Single-User DBMS* by Raima	195	**159**
db_VISTA *Single-User with Source* by Raima	495	**429**
db_VISTA *Multi-User DBMS* by Raima	495	**429**
db_VISTA *Multi-User with Source* by Raima	990	**849**
Entelekon C Function Library	130	**119**
Entelekon C Windows	130	**119**
Entelekon Superfonts for C	50	**45**
Entelekon Combination Package	200	**179**
ESP for C and Pascal by Popular Programs	200	**159**
FirsTime for C by Spruce Technology	295	**229**
GraphiC by Scientific Endeavors	250	**209**
Greenleaf C Functions Library	185	**139**
Greenleaf Comm Library	185	**139**
The HAMMER by OES Systems	195	**179**
H.E.L.P. by Everest Solutions .... *New*	395	**329**
MetaWINDOWS by Metagraphics	150	**139**
Multi-Halo *Graphics* by Media Cybernetics	250	**189**
PANEL by Roundhill ...... *Library Source Available*	295	**234**
PC Lint by Gimpel Software	139	**109**
Polytron C Library	99	**79**
Pre-C *Lint Utility* by Phoenix	395	**289**
Scientific Subroutine Library *for C* by Peerless	175	**139**

Vitamin C by Creative Programming

Powerful C library providing complete screen I/O control, field highlighting and input validation, pull down menus, user help system, extensive windowing and much, much more.
Comes with source code and no royalties.

List Price $150 Our Price **$139**

Programmer's

136 Sunnyside Street Hartville, Ohio 44632

Call For Our Catalog

We accept qualified Corporate Accounts and P.O.'s • Dealer

FOR THE IBM-PC/XT/AT and compatibles.

C UTILITIES cont.

	LIST	OURS
TopView Toolbasket by Lattice .... *New*	250	**219**
TopView Toolbasket *with source code* .... *New*	500	**429**
View Manager *for C* by Blaise Computing	275	**219**
View Manager *with Source*	425	**329**
Windows for C by Vermont Creative Software	195	**139**
Windows for Data .... *Includes Windows for C*	295	**259**

CROSS ASSEMBLERS

We carry over 25 different cross assemblers from 2500 A.D. Call us with your specific needs.

	LIST	OURS
6800 XASM by 2500 AD	200	**165**
8080 XASM by 2500 AD	200	**165**
68000 XASM by 2500 AD	300	**249**
Z-80 XASM by 2500 AD	200	**165**

FORTRAN COMPILERS AND UTILITIES

	LIST	OURS
Microsoft Fortran .... *Links with Microsoft C*	350	**229**
RM/Fortran by Ryan-McFarland	595	**439**
Btrieve by SoftCraft	250	**199**
FORLIB-PLUS by Alpha Computer Service	70	**59**
Multi-Halo *Graphics* by Media Cybernetics	250	**189**
PANEL *Screen Designer* by Roundhill	295	**234**
PolyFortran Tools by Polytron	179	**139**
Scientific Subroutine Library by Peerless	175	**139**
The Statistician by Alpha Computer Service	295	**269**
Strings and Things by Alpha Computer Service ..	70	**59**

LATTICE PRODUCTS

All products in this section have Lattice serial numbers. These products receive support and updates directly from Lattice Inc., the company that actually developed the Lattice C compiler. **Lattice C is in stock and ready for shipment.**

	LIST	OURS
Lattice C Compiler	500	**349**
C Cross Reference Generator	50	**45**
C-Food Smorgasbord *Function Library*	150	**119**
C-Sprite *Debugger* ...... *Now Supports Source Level*	175	**149**
Curses *Screen Manager*	125	**109**
Curses *with source code*	250	**219**
dBC *dBase File Manager for C*	250	**219**
dBC *with source code*	500	**429**
LMK *Make Facility*	195	**159**
Text Mgmt Utils *(GREP/DIFF/ED/WC/Extract/Build)*	120	**105**
TopView Toolbasket *Functon Library* .... *New*	250	**219**
TopView Toolbasket *with source code* .... *New*	500	**429**

MICROSOFT PRODUCTS

MICROSOFT.

	LIST	OURS
C Compiler	395	**259**
COBOL Compiler	700	**495**
Fortran Compiler .... *Links with Microsoft C*	350	**229**
Macro Assembler *w/utilities*	150	**109**
Microsoft Windows .... *New*	99	**89**
MS Sort	195	**149**
muLISP	250	**189**
Pascal Compiler .... *Links with Microsoft C*	300	**219**
QuickBASIC Compiler .... *New*	99	**79**

OTHER LANGUAGES

	LIST	OURS
Janus/ADA C Pack by R&R Software	95	**89**
Janus/ADA D Pack by R&R Software	900	**699**
Level II COBOL by Micro Focus	750	**599**
Modula-2/86 by Logitech	495	**399**
PC/Forth+ by Laboratory Microsystems	250	**209**
Professional COBOL by Micro Focus	3000	**2395**

PHOENIX PRODUCTS

Phoenix Authorized Dealer

In stock and ready for immediate shipment.

	LIST	OURS
Pasm86 *High Performance Macro Assembler*	295	**219**
Pre-C *Lint Utility*	395	**289**
Pfantasy .... *Combination of all products below*	995	**895**
Pfinish *Performance Analyzer*	395	**289**
Pfix-86 Plus *Symbolic Debugger for Plink-86*	395	**289**
Plink-86 Plus *Overlay Linker* .... *New version*	395	**289**
Pmaker *Program Development Manager*	195	**139**
Pmate *Macro Text Editor* .... *New version*	225	**159**
Ptel *Binary File Transfer Program*	195	**139**

POLYTRON PRODUCTS

	LIST	OURS
Polytron C Library	99	**79**
PolyFortran Tools by Polytron	179	**139**
PolyLibrarian *Library Manager*	99	**79**
PolyLibrarian II *Library Manager*	149	**129**
PolyMake *UNIX-like Make Facility*	99	**79**
PolyOverlay *Overlay Optimizer*	99	**79**
PolyXREF *Cross Reference Utility. . . Complete system*	219	**179**
PolyXREF .... *Support for one language only*	129	**109**
PVCS *Polytron Version Control System* .... *New*	395	**359**
PVMFM *Polytron Virtual Memory File Manager* ... *New*	199	**179**

SOFTCRAFT PRODUCTS

	LIST	OURS
Btrieve *ISAM File Manager*	250	**199**
Btrieve/N *for Networks*	595	**469**
Rtrieve *Report Generator for Btrieve*	85	**79**
Rtrieve/N *Report Generator for Btrieve/N*	175	**159**
Xtrieve *Query Utility for Btrieve*	195	**169**
Xtrieve/N *Query Utility for Btrieve/N*	395	**299**
OPT-Tech Sort by Opt-Tech Data Processing	99	**84**

TEXT EDITORS

	LIST	OURS
Brief from Solution Systems	195	**Call**
Epsilon by Lugaru .... *Multitasking*	195	**165**
ESP for C and Pascal by Popular Programs	200	**159**
FirsTime for C by Spruce Technology	295	**229**
FirsTime for MS Pascal by Spruce Technology ...	245	**199**
Pmate by Phoenix .... *New version*	225	**159**
Vedit by Compuview	150	**119**
XTC *Text Editor* by Wendin .... *Includes source*	99	**89**

TURBO PASCAL AND UTILITIES

	LIST	OURS
Turbo PASCAL by Borland International	70	**55**
Turbo PASCAL *w/8087 or BCD*	110	**95**
Turbo PASCAL *w/8087 & BCD*	125	**105**
FirsTime for Turbo by Spruce Technology	75	**69**
Multi-Halo *Graphics* .... *Offer expires Dec 10*	250	**89**
Screen Sculptor by Software Bottling	125	**99**
Turbo ASYNCH by Blaise Computing	100	**89**
Turbo POWER TOOLS by Blaise Computing	100	**89**
TurboPower Util *w/source* by TurboPower Software	95	**89**
TurboRef by Gracon Services	50	**45**
TurboWindow by MetaGraphics	55	**49**
XTC *Text Editor* by Wendin .... *Includes source*	99	**89**
Borland Jumbo Gift Pack . *Combo of all items below*	245	**219**
Turbo PASCAL	70	**55**
Turbo DATABASE TOOLBOX	55	**49**
Turbo EDITOR TOOLBOX .... *New*	70	**59**
Turbo GAMEWORKS TOOLBOX .... *New*	70	**59**
Turbo GRAPHIX TOOLBOX	55	**49**
Turbo TUTOR	35	**29**

WENDIN PRODUCTS

	LIST	OURS
Operating System Toolbox ..... *Build your own OS*	99	**89**
PCUNIX *Operating System*	49	**45**
PCVMS *Operating System* .... *Similar to VAX/VMS*	49	**45**
XTC *Text Editor* .... *Includes source code*	99	**89**

XENIX SYSTEM V BY SCO

	LIST	OURS
Xenix 86 Development System .... *For XT*	495	**449**
Xenix 86 Operating System .... *For XT*	495	**449**
Xenix 86 Text Processing Package .... *For XT*	295	**259**
Complete Xenix 86 System .... *Combined Package*	995	**899**
Xenix 286 Development System .... *For AT*	495	**469**
Xenix 286 Operating System .... *For AT*	495	**469**
Xenix 286 Text Processing Package ...... *For AT*	295	**269**
Complete Xenix 286 System ... *Combined Package*	995	**949**

XENIX LANGUAGES AND UTILITIES

	LIST	OURS
APL*PLUS/UNX System by STSC ..... *For AT only*	995	**795**
c-tree by FairCom	Call	**Call**
Microsoft BASIC Interpreter	350	**279**
Microsoft COBOL Compiler .... *New*	995	**795**
Microsoft Fortran Compiler	495	**389**
Microsoft PASCAL Compiler	495	**389**
PANEL *Screen Designer* by Roundhill ... *For AT only*	595	**539**
Windows for C by Vermont Creative Software	395	**359**
Windows for Data by Vermont Creative Software .	595	**539**

First new 1/4-inch cartridge since 1971

If you *don't* have data worth preserving, then the reasons for buying Cartrex's new 1/4-inch, high performance, virtually error-free tape cartridge won't mean anything

But, if you are one of the many 1/4-inch tape cartridge users that assume 3M's cartridges just *have* to be "good enough" for today's high performance tape drives, read these simple facts to understand why that isn't true anymore.

The new Cartrex 1/4-inch tape cartridge is the first new tape cartridge design in almost a decade and a half. Tape drive manufacturers now have a new cartridge technology which allows them to advance beyond this previous artificial barrier.

PILOTS HAVE A SAYING, "There are those who have made a wheels-up landing—and those who will." You can apply this expression to those who have lost data and those who will. Unfortunately, data loss isn't always because users haven't backed-up their hard disk. Sometimes it's because their 1/4-inch tape cartridge, where they back up their hard disk, developed hard errors—those insidious errors that tend to increase over time. That's why Cartrex has developed a 1/4-inch data cartridge for today's high performance drives that virtually eliminates errors.

Why a new cartridge

When 3M announced its cartridge in 1971, it was designed for a low capacity tape drive with less than 3 megabytes—2.88 to be exact. The tape was low in density—1600 bits per inch with only 4 tracks and 300 feet of tape.

The tolerances required for the tape drives of the early 1970's were fine for then, but today's tape drives require much tighter tolerance. Today's tape cartridges must work with drives that have 9 or more tracks and bit densities as high as 12,000 bits per inch on 600 feet of tape. That means capacity increases of 2,000 percent packed into the same cartridge.

The reasons that yesterday's cartridge technology simply won't work properly in today's high capacity drives is inherent in the cartridge design.

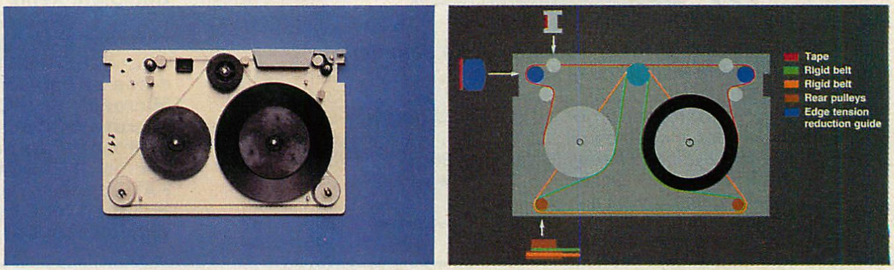
With the significant increases in capacity, the three culprits that make cartridge tolerances so important are fluctuating tape tension, redeposit nodules, and instantaneous speed variations (or ISV).

Tape Tension

Any child who has played with a magnet understands that as the magnet is separated from metal, the magnet's ability to work is decreased. So it's no surprise to find out that the closer the tape drive head is to the tape, the better the reading. This closeness is particularly important when the embedded iron filings get packed tighter in today's high density tape.

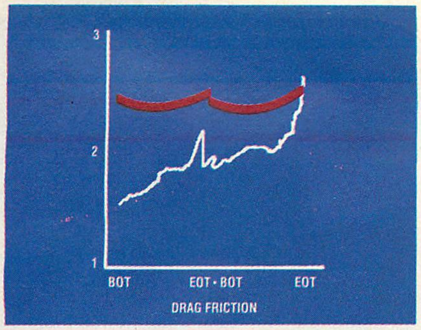
It's also important with the increase in the number of tracks. After all, you wouldn't want the head to be reading an adjacent track any more than you'd want it reading more than one magnetic representation of a bit.

Unfortunately, tape tension historically has not been constant. As the tape unwound, the tension increased. What's important is both the *amount* and *range* of tension. A fluctuating increase or decrease in tension is as unacceptable as low tension is in the first place. As the accompanying graphs



The Cartrex cartridge design (upper left) uses the N2 principle which creates head-to-tape tension by applying a mechanical differential at the rear pulley between a stiff transport belt and a stiff tensioning belt, indicated in green and yellow in the color schematic (upper right).

The conventional 3M design applies drag at the rear pulley to create tension, which creates heat, and limits future speeds much above 90 inches per second.



Good head-to-tape tension ensures the highest probability of reliably capturing data. Fluctuating tape tension allows data loss due to head-to-tape separation and smearing redeposit nodules across the tape head. The Cartrex cartridge, compared to the conventional design, creates constant and higher tension.

show, the Cartrex cartridge has higher tension and flatter profile than the 3M cartridge. This means more reliable data across the entire tape.

Redeposit Nodules

Another reason to keep constant tension is to avoid "redeposit nodules" from smearing across your tape drive's head. What are redeposit nodules? They are the insidious flakes of tape media that break off from the edges of the tape and get dragged up to the edge of the tape head. If the tension is low, or becomes low when the tape starts or reverses, the flakes come up over the edge, get smeared over the head, and reduce its ability to read the data.

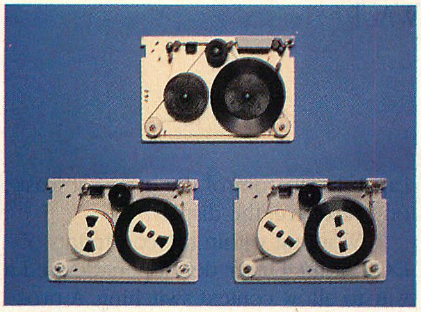
Even worse, however, is that these redeposit nodules are dragged along the surface of the tape and get embedded and packed over time. When your drive tries to read the data, the redeposit nodules act as a tent pole holding up the tape away from the head. As a result, even the best error-recognition algorithm can only tell you one thing—you've lost the data.

You might be wondering what causes the flaking in the first place. Again, it is cartridge design. The basic design uses a tape guide, shown in the accompanying illustration. The problem with this approach, is that it presupposes that the tape will always run parallel to the top and bottom caps of the tape guides. At the low speeds of 30 inches per second typical of when the 3M cartridge was designed, it was less of a problem. But at today's speeds of 90 inches per second and more, the tape wanders. When it presses against the top of the tape guide, the tape's edge pressure builds. Not only does media flake off, but you lose data due to the "coining" or "scalloping" effect.

Cartrex eliminated the cause of the tape coining or scalloping by developing a barrel-shaped roller placed prior to the tape guide. The laws of physics show that by riding on a rounded barrel, the tape will always seek the middle, reducing the tape edge pressure. This seemingly simple addition causes the tape to always enter the tape guide with zero edge pressure. In this way, the possibility of media flaking off and creating redeposit nodules is virtually eliminated.

Instantaneous Speed Variation (ISV)

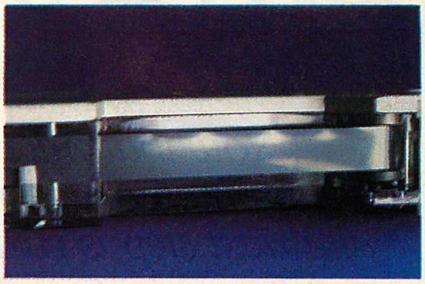
Instantaneous speed variations is exactly what it sounds like—small, instantaneous changes in tape speed as it crosses the tape head. At slow tape speeds and low bit densities—like the 1971 standard of 30 inches per second and 1,600 bits per inch—ISV wasn't as big a problem. At that time, the bits were crossing the head at 48,000 bits per second.



You can compare the Cartrex design on top with the conventional 3M design (lower left) and DEI, a 3M licensee (lower right). The omission of "edge tension reduction guides" (indicated in dark blue on the color schematic) on either the 3M or DEI design means that tape edge pressures will increase causing the magnetic media to flake off, smear across the tape head, and cause "redeposit nodules" to become embedded in the tape.

Today, however, the story has changed. 90 inches per second and 8,000 bits per inch mean that 720,000 bits cross the head every second. A 1,500% increase.

As you may have guessed, speed fluctuations in the 48,000 bits per second made reading data difficult for tape drive electronics. But when the electronics have to guess whether or not the bit rate of 720,000 bits per second is accurate, the electronics can become overwhelmed.



High speed tape without the "edge pressure reduction guide" seldom enters tape guides parallel to the top and bottom. The edge pressure which results creates "scalloping" or "coining" on the tape. The effect is data loss due to head-to-tape separation, flaking media that smears across the head, and "redeposit nodules" that create hard errors.

Never a Single Issue

Your tape drive seldom has the luxury of dealing with an isolated problem. It's usually a combination of ISV, redeposit nodules, and tension problems all together. Now you understand why Cartrex wanted to develop a cartridge for a market that needed a modern alternative.

Where to get it

Not only has Cartrex improved on the industry's 15 year old technology, they have done it for a competitive price.

Now you can have peace of mind when you back up your hard disk on today's sophisticated tape drives.

And you can have it fast. Simply phone:

(619) 485-6603 EXT 500

and give Cartrex your credit card number and they will send your cartridges by return mail.

VISA MasterCard AMERICAN EXPRESS

CARTREX CORPORATION
MARKETING HEADQUARTERS
11590 WEST BERNARDO CT. SUITE 230
SAN DIEGO CALIFORNIA 92127
619-485-6603

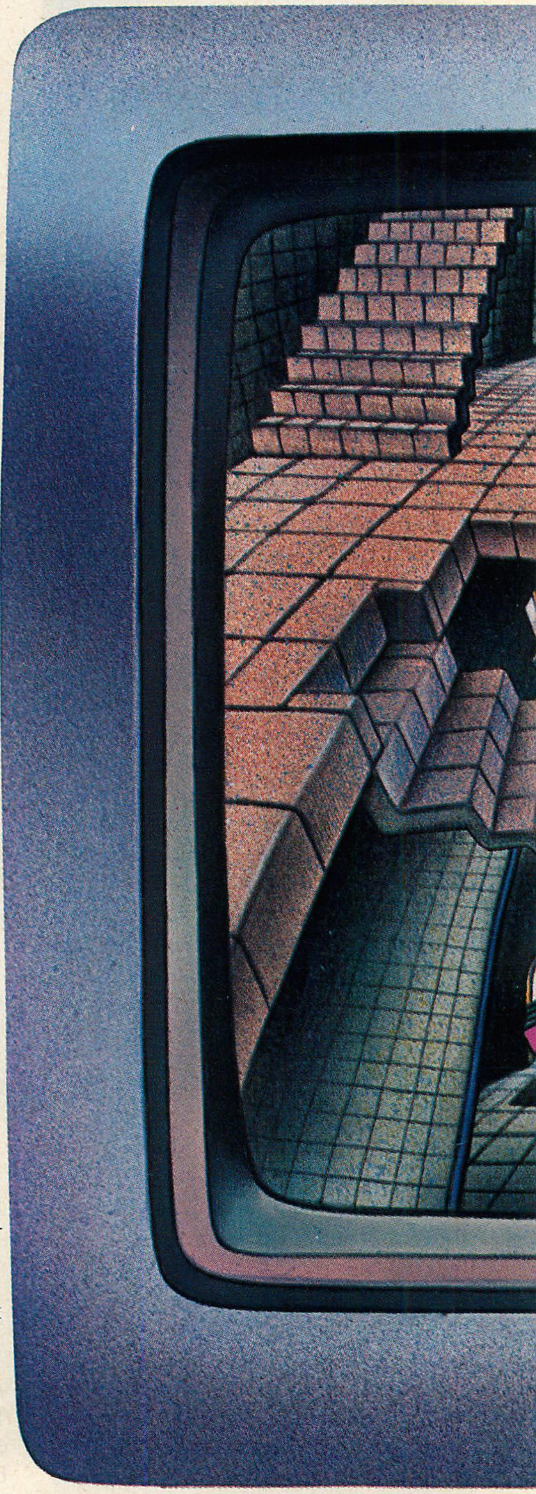
CARTREX

CIRCLE NO. 176 ON READER SERVICE CARD

Concurrent Environments

Now that microcomputers have enough memory to run multiple programs, the next step is running those programs at the same time. Eight products bring concurrency to the PC.

DON AWALT



One person, one processor, one program. This early philosophy of the microcomputer has grown outdated. While the typical computer user used to wait patiently for a program to complete, he now wants to use that waiting time more constructively. With the cost of memory plummeting, computers can be loaded with 640KB (and megabytes on an AT), giving users the space needed to run multiple programs concurrently.

Unfortunately, memory alone is not enough. The operating system must support concurrency; it must be able to control access to machine resources, including the processor, memory, peripherals, and even itself.

DOS was designed to handle only one task at a time. DOS code is not reentrant, and it has no sophisticated memory management techniques. When a program is loaded, DOS gives it all available memory with no restrictions on its accessing memory anywhere in RAM. IBM urged that all programs be written using DOS calls, but developers quickly discovered that writing screen output directly to video memory was much faster. Compatibility was sacrificed for performance, and DOS lost the capability to control the use of at least one resource, the display.

Working within the constraints of DOS, some clever developers devised a way to allow context switching. A program could insert its address into the BIOS interrupt vector table, then terminate but stay memory-resident. Another program could then be loaded and executed. The resident program would become activated when the interrupt occurred. Borland International's SideKick works by monitoring keyboard input for its Ctrl-Alt activation sequence. This, of course, is not concurrency; only one program runs at a time, and with SideKick, the user acts as the scheduler.

Eight products that bring concurrency to the IBM PC family are reviewed here. Three of the packages, DESQview by Quarterdeck Office Systems, TopView by IBM, and Windows by Microsoft, create a functional shell around DOS. Other DOS shell programs (APX Core Executive, DoubleDOS, E-Z-DOS-IT and MultiLink Advanced) provide concurrent environments with less functionality. Concurrent PC-DOS by DRI is a replacement for DOS.

Most environments offer windowing as a means of monitoring each task's progress. Some have provided a method to integrate applications through a common user interface and the capability to exchange data between applications. With advanced memory management techniques—swapping of programs to disk, support for extended memory—the DOS 640KB maximum memory limitation can be overcome. Of course, all of this must be delivered with an increase in throughput; otherwise, the reason for concurrency is lost.

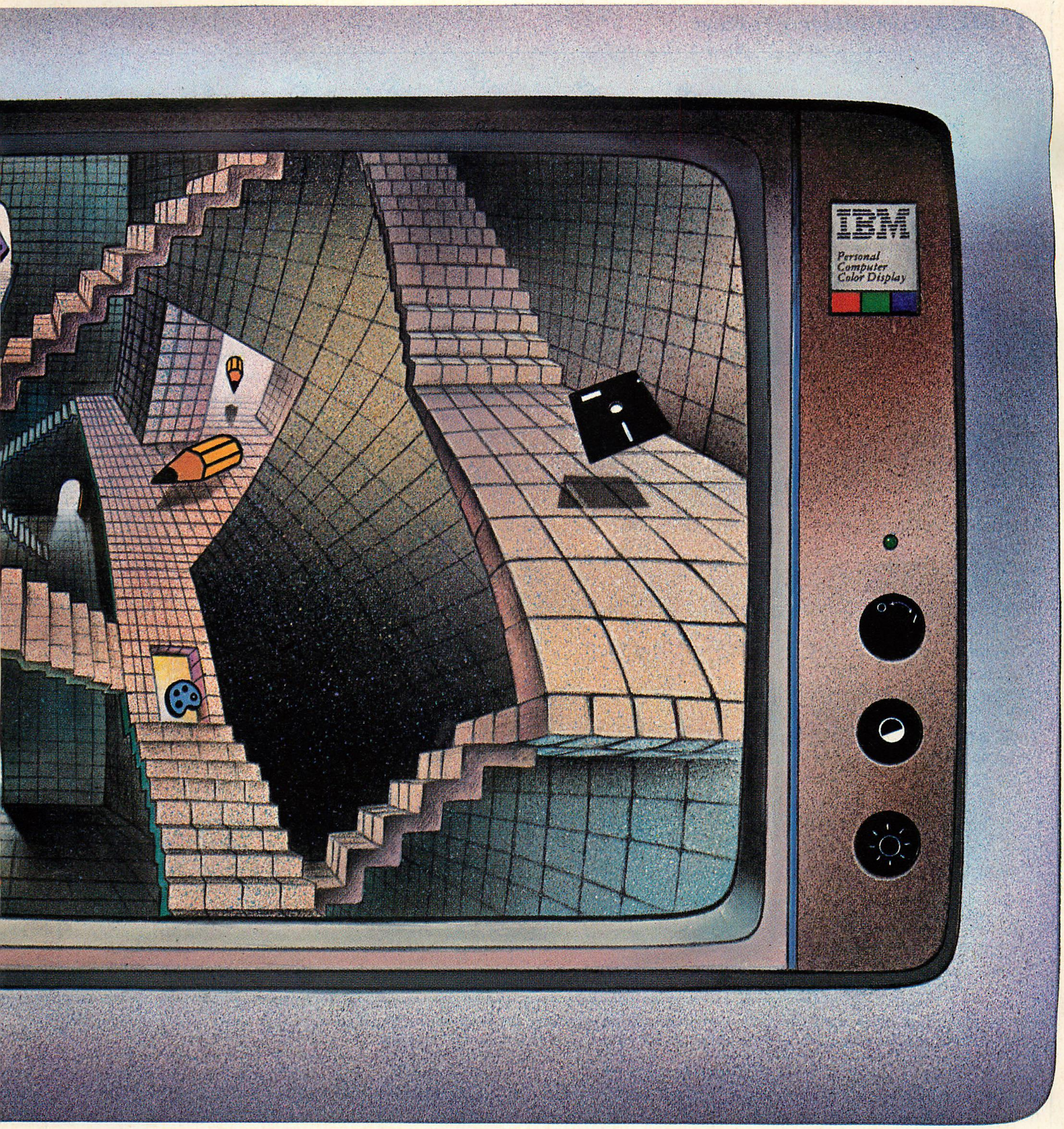


ILLUSTRATION • AKIO MATSUYOSHI

Table 1 summarizes the features of the eight concurrency products and their approaches to overcoming the barriers to concurrent processing. Most environments schedule applications programs using a time-slicing algorithm. They intercept the system clock interrupt, which fires every 55 milliseconds. The currently executing program is suspended, its operating state is stored, and another program is given control of the processor. Each program is given equal slices of execution time on a round-robin basis.

The time-slicing approach is simpler than other scheduling algorithms and can be efficient depending on the mix of applications being run. With simple time-slicing, however, a communications program may not be scheduled often enough to assure that it sees every character being received. E-Z-DOS-IT gives communications programs a bigger slice of the system clock pie as a solution to this problem. Nonetheless, time-slicing may induce unevenness in communications and result in throughputs less than the speed of the line.

The conventional approach to concurrency is that while one program is waiting for I/O to complete, another program can be run. However, access to DOS can be given to only one program at a time. While one program is executing within DOS, no other program can even begin a DOS function

TABLE 1: *Comparison of Features*

	APX	C/PC-DOS	DESQVIEW	DOUBLEDOS	E-Z-DOS-IT	MULTILINK	TOPVIEW	WINDOWS
Scheduling algorithm	Time slice	Time slice	Time slice	Time slice	Time slice, some prioritizing	Time slice or event-driven	Time slice	Non-preemptive
Memory requirements	48KB	256KB	128KB	18KB-42KB[a]	32KB	16KB	176KB-188KB[b]	142KB
Concurrent tasks	8	4	9	2	32	9	unlimited[c]	unlimited
MEMORY MANAGEMENT								
User-defined partitions	Yes	No	No	Yes	Yes	Yes	No	No
Dynamic allocation	No	Yes	Yes	No	No	No	Yes	Yes
Disk swapping	No	No	Yes	No	No	No	No	Yes
Code/data sharing	No	No	No	No	No	No	Yes	Yes
Extended memory	No	No	Yes	No	No	Yes[d]	No	Yes
USER INTERFACE								
Window support	Yes	Yes	Yes	No	No	No	Yes	Yes
Command line interface	Yes	Yes	No	Yes	Yes	Yes	No	No
Mouse support	No	No	Yes	No	No	No	Yes	Yes
Expanded keybd. buffer	No	No	No	Yes	No	No	No	Yes
Macros	Yes	Yes[e]	Yes	No	No	Yes	No	No
HANDLING OF DIRECT VIDEO MEMORY ACCESS								
Auto/Manual suspension	Manual	Both	Both	Both	Auto	No	Both	Auto
Applications patches	No	No	No	Yes	No	Yes	No	No
Monitor programs supplied	Yes	No	No	No	No	Yes	No	No
OTHER FEATURES								
PIFs	No	No	Yes	No	No	No	Yes	Yes
Cut/Paste	Yes	No	Yes	No	No	No	Yes	Yes
Print spooling	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
MS-DOS support								
Version 2.x	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Version 3.x	Yes	No	Yes	Yes	No	No	—[f]	Yes
Developer's toolkit	Yes	No	No[g]	No	No	Yes	Yes	Yes
Bundled applications	None	See text	See text	None	None	See text	See text	See text

[a] *Memory requirements are based on monitor type and menu type*
[b] *Depending on DOS version*
[c] *Limited by available memory*
[d] *Custom extended memory specification*
[e] *Function keys only*
[f] *TopView runs under DOS 3.x but cannot run programs that make DOS 3.x specific function calls*
[g] *The TopView toolkit can be used with DESQview*

Although most of these eight concurrent environment products use the same scheduling algorithm, they differ widely in memory management, window support, and the ability to run ill-behaved programs concurrently.

call. Because disk I/O is accomplished through DOS function calls, no other programs that require DOS services can execute while another is waiting for the I/O to complete. APX Core Executive and DoubleDOS manage this problem by breaking large I/O requests into many smaller requests, so that scheduling can occur more often. TopView intercepts DOS function calls and emulates their execution in order to provide greater access to DOS functions.

Microsoft Windows uses a non-preemptive scheduling algorithm. A program runs without interruption until it relinquishes control by issuing a DOS function call. Processor-bound programs can thus monopolize the machine, but with most mixes of programs this is an efficient algorithm.

Environments use two types of user interfaces—either the DOS command line or a custom interface. The DOS command line interface is created by running a copy of COMMAND.COM as a daughter process of the environment program for each applications program. This interface is easy for experienced PC users, because the DOS command line is familiar; novices, however, still need to learn DOS commands and syntax. More advanced functions, such as windowing and data exchange, are not usually available.

Table 1 lists the number of concurrent tasks supported by each package. Most have a specific limitation, although TopView can run as many applications as fit in memory, and Windows can run virtually an unlimited number.

MANAGING MEMORY

Memory management varies for each environment. Some environments require that memory be divided into

fixed-size partitions and that operating characteristics be assigned to each partition. Each partition is then available for running one application according to the predefined characteristics of the partition. The environments that use partitioning require a smaller amount of memory compared to other environments. Table 1 shows the memory requirements of each environment excluding DOS, applications, and memory requirements that vary based on the number of concurrent applications.

Other environments allocate memory dynamically. These use PIFs (Program Information Files) to describe the characteristics and needs of an applications program. Information files describe the amount of memory required by a program, operating characteristics (peripherals directly accessed, method of video I/O, interrupts processed), and any other specific information required by the environment to control the execution of an application.

Disk swapping is used by some environments to run more programs than can fit in memory at one time. Suspended applications are moved to disk and recalled later when activated or when memory becomes available. The limiting factor on the number of applications that concurrently execute then becomes performance.

Extended memory can stretch the number of programs that may be run concurrently. Memory beyond 640KB can be used for swapping programs. MultiLink Advanced supports a custom extended memory management specification; DESQview and Windows support the Lotus/Intel specification.

Code/data sharing is supported by several environments; it optimizes system memory usage. Multiple copies of the same or different programs share code segments (copies of the same program) or data segments (where simultaneous usage is managed.) An environment using code/data sharing executes multiple programs with a minor increase in used system memory.

Defining foreground and background tasks is a method for managing access to the display and keyboard. The foreground task is given control of the keyboard, and in many cases, the entire screen. There is one foreground task; all others are background tasks.

WINDOW SUPPORT

Windowing allows multiple applications to share the screen. Either with or without the application's knowledge, the screen can be partitioned into tiled or overlapped segments called *windows*, with each one displaying a portion of an application's output. One application can own multiple windows. The user can size, move, or hide windows, and scroll data within a window.

Most environments that provide the DOS command line interface do not support windows. These environments require the user to switch a program to the foreground to view display output. Although this implementation minimizes compatibility problems between applications programs that access video memory directly, no easy method exists to determine when a background program has completed.

The greatest problem for environments is proper execution of applications that perform direct video I/O—directly writing screen images to the PC's video memory, performing I/O to video hardware circuitry, or otherwise bypassing DOS. When an applications program writes directly to video memory, it interferes with the environment's control of windows. Applications that do direct video I/O are called *ill-behaved.*

A final solution to ill-behaved video I/O is a monitor program that executes the applications software interpretatively.

Table 1 summarizes ways to execute programs that directly access video memory. One method automatically suspends an ill-behaved program whenever it does not have full control of the screen. When the ill-behaved program is executing in full-screen mode and a second program is started, the first program is suspended until full-screen operation is resumed.

Suspension also can be performed manually by the user. As multiple applications are run, the user may see that programs are competing for the display in a disruptive manner. This is not an optimal method because the burden is on the user to prevent corruption of the screen.

Another strategy is to patch standard applications software to prevent the application from writing to the display in a nonstandard manner or to redirect output to an internal display buffer that is then mapped to physical video memory by the environment. Environments using this feature supply the patches for supported applications.

Patching is not an optimal method for providing compatibility, because after a program has been patched, the manufacturer's warranty support may not be available. Patches also are tied to a single version of software; when a new version of software is released, the user requires a new patch.

A final solution to ill-behaved video I/O is a monitor program that executes the applications software interpretatively. The user runs the monitor program with the applications program's name as an argument. The monitor program controls execution and redirects or prevents I/O from disrupting the display. Monitor programs slow the execution time of applications and conflict with copy-protected software, but they provide another attempt to render ill-behaved programs compatible.

Any ill-behaved application creates problems in a concurrent environment. A well-behaved program should access the screen or keyboard through DOS function calls. It should not address absolute memory locations or assume that all memory is allocated to it. Program loops should not be used to time applications, and an application should not terminate and stay resident.

Mouse support is a major benefit of environments. The user can quickly pull down a menu, move the mouse to point the cursor at a target command, and give it a click. A mouse streamlines the use of other environment features as well. Window management (scrolling, sizing, and moving windows), for example, is faster with a mouse than through a keyboard.

Several features can improve the effectiveness of the keyboard, however. Expanded keyboard buffers increase the number of keystrokes that can be typed ahead. Macros permit the user to assign a multiple keystroke sequence to a single key. The macros offered by each environment have varying capabilities; in some cases, the macro features rival the functionality offered by stand-alone keyboard macro programs.

Cut and paste is the phrase that usually describes the data exchange between applications. Cutting and pasting is difficult to implement because the data cut from one application's screen must be converted into the receiving application's data format and sent to the receiving application as input.

Several environments provide for cutting and pasting text only. The more difficult task is cutting and pasting graphics between applications. The best

job is done by Windows, which supports cutting of graphic regions from any application. Pasting can be done only into a WinAp (Windows application), a program that uses Windows function calls for system functions and is completely Windows-compatible.

Most environment packages support print spooling, or background printing of documents from multiple programs. Some use the DOS PRINT command to implement spooling; others provide their own print spoolers.

Table 1 lists other applications bundled with each environment. These bundled applications provide needed desktop management tools, which help to justify the expense of the environment product. A large number of applications are bundled with Windows.

Table 1 also indicates the versions of DOS supported by each environment. Some environments do not yet provide support for DOS 3.0/3.1; although TopView runs under DOS 3.x, it cannot run any program that makes DOS 3.x-specific system calls. This makes even DOS programs such as FORMAT incompatible with TopView.

One more important item listed in table 1 is the developer's toolkit. Many environments provide technical information on the function calls to make in order to use the features found in the environment. TopView and Windows especially provide a great deal of technical information on a rich set of functions. Others provide function calls to do a small number of tasks.

Table 2 reveals interrupts monitored by each environment. A program may take or release interrupts as different processing occurs, causing this list to change, so this information represents only an approximation of the program's control over the PC.

Microsoft Windows intercepts few interrupts; it controls the execution of applications by monitoring Windows function calls or through direct control of the DOS function and program terminate calls. As expected, Windows has intimate knowledge of the DOS program and data areas and probably uses that information to maximum effect.

The DOS command line interface programs, such as APX Core Executive, DoubleDOS, and E-Z-DOS-IT, have a large number of interrupts. Control of interrupts is tighter because these packages are less intimately tied with DOS.

TABLE 2: *Interrupts Controlled*

INTERRUPT	FUNCTION	APX	CPC-DOS	DESQVIEW	DOUBLEDOS	E-Z-DOS-IT	MULTILINK	TOPVIEW	WINDOWS
0	Divide By 0	x	x	x	x	x			
1	Single step		x			x			
2	Nonmaskable		x						
3	Breakpoint		x		x	x			
4	Overflow	x	x		x				
5	Print screen	x					x	x	x
6			x						
7			x						
8	Time of day	x	x	x	x		x		
9	Keyboard	x	x		x				
A			x				x		
B	Comm	x	x		x	x	x		
C	Comm	x	x				x		
D	Fixed disk								
E	Diskette		x						
F	Printer		x				x		
10	Video I/O	x	x	x	x	x	x	x	
11	Equipment check								
12	Memory size		x	x	x				x
13	Diskette I/O	x	x	x	x	x		x	
14	RS-232 I/O	x		x	x		x		
15	Cassette			x				x	
16	Keyboard	x	x	x	x	x	x	x	
17	Printer	x	x		x		x	x	
18	Resident BASIC								
19	Bootstrap		x	x	x	x			
1A	Time of day				x				
1B	Keyboard break	x	x	x	x	x	x	x	
1C	Timer tick		x		x			x	
1D	Video initialization								
1E	Diskette params		x						
1F	Video characters		x						
20	DOS program term	x	x	x	x	x		x	x
21	DOS function call	x	x	x	x	x	x	x	x
22	DOS term address	x	x	x	x	x		x	x
23	DOS Ctrl-Break exit	x	x	x	x	x		x	
24	Fatal error exit	x	x	x	x	x		x	
25	DOS abs disk read	x	x	x	x	x			
26	DOS abs disk write	x	x	x	x	x		x	
27	DOS terminate	x	x	x	x	x		x	

An environment may change the interrupts it controls depending on the current state of the machine, but this information indicates its control over the PC.

EIGHT ENVIRONMENTS

The features of each of the eight concurrent environment products are discussed individually and in detail below, beginning with the newest, and perhaps the most exciting, entry in the market, Microsoft Windows.

Windows. This product provides a Macintosh-like graphics interface, adds multitasking capability, window support, data interchange capability, and support for expanded memory. At the time this was written, Windows had been released only to developers, but a retail version should be available by the time this article is published.

Windows requires a PC running DOS 2.0 or later with two disk drives (either one diskette and one fixed or two diskette drives) and at least 256KB of memory. To take advantage of the true power of Windows, however, a user should have 512KB to 640KB of memory. Either a monochrome or color monitor can be used, but a graphics adapter card is required. Devices supported in the initial release include the IBM Color Graphics Adapter, Her-

cules Graphics Card, and all IBM Enhanced Graphics Adapter modes. Only the IBM EGA allows the use of color.

A mouse is a helpful feature to use with Windows, but it is not required; the package also comes with an excellent keyboard interface. A 12-lesson tutorial walks the user through various techniques for getting optimal use out of a mouse; other topics covered include dialog boxes, pull-down menus, scroll bars, and window sizing.

Windows performs well. This can be attributed to its being written as a close partner to DOS (who could do that better than Microsoft, the developer of DOS). Also, Windows uses a non-preemptive multitasking algorithm; application tasks are rescheduled only when a call to DOS is made.

Two methods are used for displaying a WinAp's output in less than a full screen. The first is scaling, best demonstrated by the clock program supplied with Windows. As the clock is placed in smaller windows, the full clock is scaled to fit in the available space. Although the clock is drawn to the same logical coordinates, the mapping of the image to the physical screen changes as the window shrinks. Scaling is provided by the Windows environment; the application need follow only a few rules.

The second method requires the application to take full responsibility for its display. The Windows environment sends messages to the application, informing it that its window has changed in size or must be repainted. The application must be prepared to deal with these messages at any time. The MS-DOS Executive section of Windows (which is itself a WinAp) is a good example. As the window changes in size, the program rewrites the file list in more or fewer columns of shorter or longer length. If the window is too small for all the information, the DOS Executive adds scroll bars to indicate that more information is available outside the boundary of the window.

Macintosh features abound in this product. Dialog boxes are used by the system whenever the user needs to be prompted for additional information. They optionally include five features: a list box, text box, option buttons, check boxes, and command buttons. A list box displays information for user selection (for example, a list of files). A text box allows the user to type in information (for example, a file name). Option buttons (for one choice out of many) and check boxes (for several choices out of many) allow the user to select options. Command buttons are for action; clicking on them usually denotes that the user is ready to proceed.

Windows places an icon at the bottom of the screen for each task not currently shown in a window. When the application is a WinAp, the icon is a picture that describes the application (for example, a clock or calculator). In other cases, the icon is a square that displays the first three letters of the program title (as described in the PIF).

When icons are created in the work area, they can be used to bring an application back to share the screen. The cursor is placed over the icon, and the mouse is clicked. The icon is dragged to a border of the screen, and the mouse button is released, causing the new application to share a portion of the screen. Note that well-behaved programs in icon form in the work area are still executing.

Macintosh-like features abound in Windows. Dialog boxes are used by the system whenever the user needs to be prompted.

Zooming an application (making a window the full size of the screen) is done by placing the mouse cursor in the size box located at the top right-hand corner of the window. A double click at this position zooms the window to full-screen size.

Shrinking a window can be done when the user is temporarily finished viewing the application. A double click of the mouse while the cursor rests within the title bar at the top of the window turns the window into an icon in the work area.

The System menu box (described below) is located at the top left corner of the window; by double clicking the mouse while the cursor is located within the system menu, an application is closed (terminated).

Other features of Windows include support of Microsoft's graphics device interface (GDI) graphics primitives and GSS's (Graphics System Software) VDI drivers. The full 256 raster operations are supported; raster operations use combinations of a source bit map, pattern bit map (used as a mask), and destination bit map with combinations of Boolean operations to produce desired pixel color results. The four different Boolean operators are BIT OR, BIT EXCLUSIVE OR, BIT AND, and BIT NOT. Windows is the only graphic interface, including Macintosh, to support all 256 operations. This device independence shows Microsoft's intention to support as many graphics products as possible. Microsoft is soliciting graphics drivers from hardware manufacturers and will undoubtedly include them on distribution disks to dealers or provide them through their CompuServe forum; hardware manufacturers will likely distribute drivers for their products.

The Clipboard is the repository of data that are cut from applications for use in pasting into applications. Even ill-behaved applications can use the Clipboard; Windows provides the unique capability to copy any screen of data to the Clipboard when Alt-PrtSc is pressed. Clipboard data can be pasted, printed, or just viewed.

Several menus of commands provide additional features in Windows. For example, the System menu is a pull-down menu that contains commands for manipulating applications windows. These windows can be sized, moved, turned to icons in the icon work area, zoomed, and closed.

The DOS Executive has three menus: File, View, and Special. The File menu provides file-oriented commands: Run, Load (which runs an application in the background with its icon in the icon work area), Copy, Get File Information, Delete, Print, and Rename.

The View menu enables the user to select criteria for viewing a directory listing. Short and long listings are available; the short listing shows file names only, the long one is similar to DOS DIR. In addition, all files can be displayed or a partial listing sorted alphabetically, by date, size, or extension.

The Control Panel appears when the program CONTROL.EXE is run. This window has three menu offerings: the Installation menu (for adding/deleting a printer), the Setup menu (for changing the printer port, specifying printer type, and setting parameters for communications ports), and the Preferences menu (for setting screen colors for backgrounds, borders, etc.).

Windows runs programs in five different modes. The highest level of functionality is a WinAp. This program can take advantage of the full set of Windows features, including concurrency, mouse/cursor use, the Macintosh-like interface, and data interchange.

The next level of functionality is an "old" application that can run in a

window and execute concurrently. This program type does not directly modify the video memory, keyboard, or absolute memory addresses and is not a terminate-and-stay-resident program. Sufficient system memory must exist in order to execute the program.

The third level is an application that runs using the whole screen, with the user being able to switch back to Windows. This application does not directly modify the keyboard or memory. Sufficient system memory exists to execute the program, and the application does not prevent the switch. No other program can run concurrently with this program. Turbo Pascal and WordStar are examples of this type of program.

At the fourth level, the applications program does not directly modify memory and sufficient system memory exists to execute the program. However, Windows effectively goes to sleep. The user cannot switch from the application to Windows. When the application terminates, Windows wakes up.

At the lowest level of functionality, Windows releases all but a slice of its memory and virtually hands the machine to the application. When the program has completed, Windows reloads and restarts.

Microsoft recommends running programs that terminate and stay resident *before* Windows is run; in any case, SideKick steals interrupt vectors based on its own internal timers and is, therefore, incompatible with Windows. BASICA is an ironic thorn in Windows' side, just as it is to the other concurrent packages. Exiting BASICA kills the serial mouse, because BASICA reprograms the serial ports (despite any tricks Windows plays to convince the system that the system has no serial ports).

Additional programs are bundled with Windows. Notepad is an electronic memo pad, similar to the function provided in SideKick. A calendar, calculator, clock, and a game called Reversi are also included. Cardfile is a graphic representation of a set of index cards used for limited database operations; a similar function is provided with Concurrent PC-DOS.

One program that was not ready for release in the premier version of Windows is a communications program that connects to remote or local computers. Attempts to communicate with a DEC PDP 11/44 at 9600 baud resulted in numerous dropped characters and lines. Microsoft is aware of this bug, which arises from interrupts being disabled too long, and expects the bug to be corrected in the final version.

The Windows Toolkit is available for $500 to developers and offers descriptions of window functions, GDI support, memory management, system resources, raster/graphic operations, and font support (Windows supports many fonts). Microsoft is making a run-time version of Windows available to third-party developers, so that packages can be developed that use the basic multitasking/windowing/interface features of Windows available for a stand-alone product that runs without Windows. The retail version will cost approximately $99.

TopView. IBM's TopView, version 1.01, is a concurrent windowing package that can run on the entire PC family with the exception of the PC*jr*. A minimum of 256KB memory is required, although 512KB is recommended. Also required are two disk drives (one diskette and one fixed or two diskette drives) and a monochrome, color, or enhanced color display. The 3270-PC requires a minimum of 448KB memory, two disk drives, monochrome or 3270 Color Display, and an APA Adapter for running graphics applications. TopView runs under DOS 2.0, 2.1, 3.0, or 3.1, but does *not* support programs that use DOS 3.x specific system calls. Testing revealed problems with DOS commands FORMAT, LABEL, PRINT, RESTORE, RECOVER, SELECT, SORT, SYS, TREE, GRAFTBL, GRAPHICS, RMDIR, and MODE. Batch files, redirection, and piping are not supported.

The strength of Microsoft Windows in features and future support creates a worthy contestant in the concurrent environment market.

Applications concurrently execute through one or more windows. A TopView window has two characteristics: coordinate dimensions (the size and position of the window on the PC display screen) and foreground/background status. Applications use a TopView-managed logical display for display I/O; TopView maps data from the full-sized logical display to the application's window in the physical display. For example, a window encompassing the left half of the display shows the left half of text lines in the application's logical display. Windows can be overlapped or tiled, at the user's discretion. The foreground program's window is framed by a double border.

Programs that abide by the rules of DOS compatibility (no writing directly to video memory, proper memory management) can execute concurrently in a window. TopView's task scheduling algorithm gives applications running in the foreground a larger share of the PC's processing time.

A PIF is created to provide TopView with information about an application. When adding a program, the user supplies a title, path name, and memory requirements. Other information in the PIF is given default values and may be changed using the Change Program Information function. A PIF built on these defaults may cause one of two problems: the assumptions may be too conservative and the program will not be able to take advantage of many of TopView's features (such as windowing and background execution); or the program will not work at all. In either case, some experimentation with the PIF is a worthwhile exercise.

The Change Program Information function is available through the Programs selection on the TopView menu. The user enters command line arguments for the program (TopView assumes the program takes no arguments), path to data files (program and data are assumed to be in the same directory), minimum, maximum, and system memory requirements, screen type, window size, video pages used, display position of the window, shared program and data path name, and whether the program writes directly to video memory. The user also indicates whether the program directly accesses the BIOS keyboard buffer, can run in the foreground only, and uses the 8087/287 math coprocessor (so that TopView will save the 8087/287 register state when task switching). The PIF is the only way to enter command line arguments. This means the user must edit the PIF every time the program is invoked with different arguments.

Entering appropriate values for minimum/maximum/system memory may make an ill-behaved application run well under TopView. The system memory PIF entry is 7KB if the application uses text and 23KB for graphics.

Row and column dimensions of windows are defined in the PIF. A user, therefore, can arrange the window definitions so that favorite programs are automatically placed on the screen in a customized arrangement. The PIF also

includes information on the range of interrupt vectors that are controlled by the applications program.

TopView provides the ability to share a single copy of code or data between multiple applications. This feature requires programming the applications software. Sharing integrates several software packages or multiple copies of the same software package.

The TopView menu, displayed by pressing the Alt key or mouse button 3 (both buttons pressed on a two-button mouse), provides a selection of the major program features. Scroll allows the user to scroll text within a window to see hidden data. The Window option is used to define the dimensions of the application's window, and its selection produces another menu with options Move, Size, Zoom, and Unzoom.

Data exchange capability between different applications is provided through Scissors. Windows that are in text mode (currently, graphic applications do not run in windows), share textual data through three Scissors functions: Cut, Copy, and Paste. Applications that access video memory directly are less well-behaved and may not be able to support Scissors functions.

TopView has created a unique Filter function, supported by the software manufacturer, to enhance applications that are not TopView Aware. As an alternative to modifying an application to be TopView Aware, Filter can be used to translate mouse movement to the program's normal keyboard input. For example, when the mouse moves, the program can be sent its cursor movement keys. It defines which keystrokes should be sent to the program for starting a Cut, ending a Cut, beginning a Paste, and ending a Paste. Other translations are provided in the Filter table. Many manufacturers already supply filter tables, providing TopView compatibility with minimal effort.

The Suspend function stops an application in the selected window from executing. Suspended programs are hidden until they resume execution.

Switch provides a menu of programs for selecting a new foreground task. A legend is displayed to the left of each program name when the program is hidden or suspended. When multiple copies of the same program are running, TopView assigns a program number along with the program name. Running multiple copies of the same program must be done with care; those that do not create unique names for their temporary files cannot be run concurrently from the same directory.

When the Programs function is selected, a second menu appears with these options: DOS Services, Add or Delete a Program, and Change Program Information. DOS Services presents a menu of basic DOS functions: COPY, PRINT, TYPE, RENAME, ERASE, and OTHER (OTHER is chosen to select a DOS function besides those described on the menu; the command name is typed along with its arguments). In addition, a second window appears that shows files in the current directory, and a third menu enables the user to choose directory sorting by Name, Extension, Size, or Date/Time. DOS Services is an example of one application using three windows (one application can manage multiple windows).

The final three TopView functions are straightforward: Help provides context-sensitive assistance to TopView functions; Quit stops an application; and

TopView has created a unique Filter function that can be used to enhance any applications that are not TopView Aware.

Exit leaves the TopView environment and returns to DOS. The Exit function requires a clean environment—that is, all programs terminated.

Several bundled programs included with TopView provide additional desktop functionality and serve as examples for performing TopView functions: an alarm clock, calculator, and a color change program.

The package comes with an interactive tutorial and documentation guide. Documentation is written clearly and uses pictures and examples.

TopView is not without problems; the most notable is the lack of graphics support in windows, a severe disadvantage when compared to Microsoft Windows. Some benchmark tests show significant degradation under TopView.

TopView manages memory inefficiently. Disk swapping is not used; all running programs must fit in available memory. Enhanced memory beyond 640KB is not supported. When an application is started, it is loaded into the first available memory area large enough to fit the program. TopView adds programs in this manner until memory is filled. When a program completes, its memory is freed for other programs, but noncontiguous memory areas are not combined. Sufficient memory may be available to run an application, but the lack of a contiguous block of memory will prevent its start. Furthermore, use of multiple windows or the Scissors option from within an application requires additional memory to hold the logical displays and the cut area (called a Clipboard).

TopView is not compatible with programs that terminate and stay resident. The standard DOS debugger is not supported; IBM recommends the RDT debugger in IBM's Professional Debug Facility package ($100). In addition, TopView does not allow the diskette drive on an XT to be addressed as drive B: as well as drive A:. Attempts to do so result in the error message "Invalid drive specification."

APX Core Executive. Core Executive, version 2.1, from Application Executive Corporation, supports as many as eight simultaneous tasks in tiled or overlapping windows. Each task includes a separate running copy of COMMAND.COM for each user-defined partition. Core Executive can be used with a monochrome or color display and requires only one disk drive (which can be either floppy or hard).

The product provides a basic set of commands to support concurrent execution. SWAP TASK switches tasks between foreground and background. Foreground tasks have control of the screen and keyboard. Because background tasks cannot update their windows until they are switched to the foreground, it is difficult to determine when a background task is completed. TRANSCRIBE provides cutting and pasting of text data. WINDOW creates, sizes, and moves windows to a desired screen location. Cursor keys scroll data in a task's window to view areas not currently appearing on the screen.

Core Executive provides the Keysave macro function for defining 16 keyboard macros with 1,024 keystrokes. Keyboard macros can invoke other keyboard macros, up to 16 levels deep, and can be used for applications or for Core Executive commands.

Two utilities are supplied with Core Executive. APXUTIL saves Keysave definitions to disk, enables and disables concurrent processing (the default is DISABLED), enables and disables windowing, and redirects printer output to serial ports. Programs that write directly to screen memory are *windows disabled*; they cannot be run in a window.

The second utility, APXRUN, is a monitor program that can run an ill-behaved application. Programs that are run by APXRUN are prevented from disrupting the display and other programs; they include BASICA, BASIC, Crosstalk, Edix, Lotus 1-2-3, Multiplan, pfsfile, pfswrite, SuperCalc, SuperCalc 2, VisiCalc, and WordStar.

APX documentation provides a tutorial to guide the user through each Core Executive command, a list of error messages and their causes, and a program interface guide with technical information for developers.

Several limitations compromise Core Executive. The DOS emulation is not complete; DOS editing keys are not supported or provided through the Keysave function; and the DOS PRINT command is not supported. BREAK is always OFF, so programs cannot be terminated by Ctrl-Break. Rebooting is the only way to terminate Core Executive operation.

Concurrent PC-DOS. Concurrent PC-DOS (C/PC-DOS) version 4.1 by Digital Research is an operating system with concurrent characteristics. C/PC-DOS coexists on disk with DOS, supporting the execution of CP/M and DOS programs. (For a detailed review of C/PC-DOS features in addition to those that pertain to concurrent environments, refer to "Concurrent PC-DOS," Don Awalt, *PC Tech Journal*, March 1985, p. 45.)

The recommended configuration for using C/PC-DOS is a PC/XT, PC/AT, or AT&T 6300 with 512KB memory, one or two diskette drives, and a hard disk. Because disk swapping is not used, the more memory, the better.

The major enhancement of C/PC-DOS 4.1 is DOS 2.0 path-name support. Unfortunately, use of DOS programs is not recommended because they may take advantage of internal structures not accessible from C/PC-DOS. This compromises the reality of C/PC-DOS's compatibility with DOS.

Windows are easily sized and moved, and color attirbutes can be set for each. Switching between windows is a two-keystroke process, Alt-tasknumber. If this key combination conflicts with an application's own keyboard usage, other keystrokes can be defined by the user for window switching.

Multiple applications writing directly to video hardware may result in a corrupted screen. C/PC-DOS does a good job getting ill-behaved background tasks to execute, however. One notable exception is BASIC; multiple copies do not execute concurrently because each one tries to access the same BASIC workspace in low memory.

C/PC-DOS does not offer cut-and-paste capability, DOS 3.x support, or virtual or extended memory usage above 640KB. One advantage of C/PC-DOS is its ability to run existing programs without an information file.

C/PC-DOS's spiral-bound user's manual is easy to read and includes illustrations of frequently used screens. It has both tutorial and reference sections to satisfy users of varying sophistication.

DESQview. DESQview 1.0, by Quarterdeck Office Systems, offers a concurrent windows environment similar to, and in many ways better than, IBM's TopView. This product is targeted to frequent users of Lotus 1-2-3 and word processors, judging by its examples.

It requires at least 256KB; recommended memory size is 512KB (DESQview itself is 128KB). The product supports monochrome and color monitors, and screen updates are synchronized with vertical retrace (to eliminate snow on some monitors) or done at any time (for faster screen updates). A hard disk or a RAM disk for task swapping are recommended.

C/PC-DOS does not offer cut-and-paste capability, DOS 3.x support, or virtual or extended memory usage above 640KB.

DESQview is rich with features that are reached through an extensive set of menus. Multiple copies of the same program run simultaneously. Disk space or virtual memory space greater than 640KB is required to swap out suspended programs. The user can command DESQview to swap a program to disk in order to suspend it or to make room for a new program.

DESQview is written emulating TopView's applications program interface, and runs "TopView Aware" programs (those that make TopView function calls) from within a window.

Window support includes the ability to change size and position, hide a window, zoom to full screen, or shrink to original window size. Windows can be as small as one line by one character. To simplify window management, the user can choose from nine screen positions with a unique color combination for each window.

DESQview supports graphics that can scale bit-map graphic displays (a feature not supported by TopView). Scaling is slow, and any poorly behaved program that writes directly to a screen is suspended when scaled (and activated when returned to full size.)

The Mark/Transfer function is used for cutting and pasting. Mark selects text or numeric data, single or multiple lines, or complete files. Mark/Transfer reformats the data of popular business programs, for example, data from dBASE III record format can be changed to WordStar text format. Mark/Transfer can be used with Lotus 1-2-3, pfsfile, and Crosstalk, among others.

DESQview enhances the traditional set of DOS services. Directory listings are sorted on two levels using eight different criteria. The most commonly used DOS commands (BACKUP, COPY, ERASE, PRINT, RENAME, and TYPE) are available on a primary menu; additional DOS commands are available on a second menu. This set-up eliminates tedious scrolling through a large list of commands. DOS services also can be run in a background window.

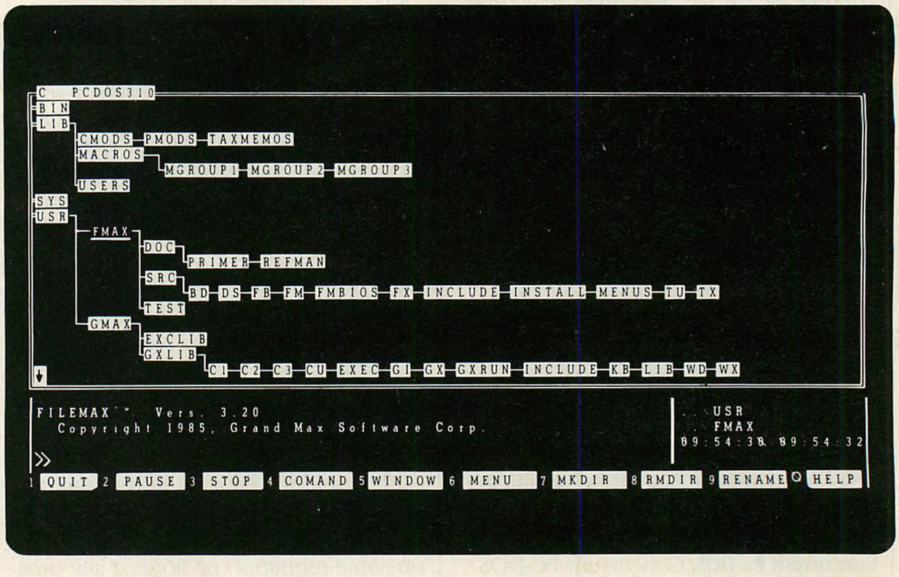
DESQview adds a facility for programs that do not accept path names for data files. A path name can be assigned to a logical drive name. For example, logical drive D: can be assigned to the path C:/test/testl. Then a program such as dBASE II, which does not support paths, can access C:/test/testl/file.dat as D:file.dat.

Programs must be installed to run in a DESQview window. (Some well-known programs are preinstalled.) Two methods are used for installation. A TopView PIF can be used; or information can be given for building a DVP (DESQview PIF): name, keystroke sequence for opening the window, command to start the program, memory requirements, and program path name.

DESQview assumes that programs and data files are stored in the same directory, that programs write to the full screen (and may not run in background), can be swapped to disk (that is, they are not memory-resident), do not perform bit-map graphic operations, are "TopView Aware," choose that their window is to be automatically closed when execution ends, use their own color combinations, permit keyboard type-ahead, and run off of a diskette (and prompt the user with "insert diskette" when necessary).

DESQview is incompatible with programs that write directly to screen memory or to video hardware; these programs run in full-screen mode.

The more files you use, the more you need **FILE SYSTEM MANAGEMENT . . .**



You can't manage what you can't see; FILEMAX lets you visualize your file system while you work on it.

FILEMAX™

THE DEFINITIVE BACKUP AND FILE MANAGEMENT SYSTEM FOR HARD DISK USERS.

We changed our product name from FILEMASTER to FILEMAX to end the confusion: FILEMAX is unique. There is no other PC FILE SYSTEM MANAGER which offers the quality and range of file management features you get with FILEMAX.

FILEMAX' fast and versatile backup program is only part of the story. File re-structuring, directory sorting and renaming, and disk optimization (eliminates data fragmentation), are just a few of the powerful file system management capabilities that are all under one roof with FILEMAX.

The new FILEMAX release also includes an expanded menu system, new file recovery capabilities, and an improved user interface that lets you 'look ahead' as FILEMAX is processing your files.

FILEMAX is a state of the art file backup utility, plus a complete set of tools for managing the complexities of the DOS file system.

You may only use FILEMAX a few minutes each day; but those few minutes will pay dividends all day long.

FILEMAX™ is now available for the IBM PC-XT, PC-AT, and DOS versions 2.0 through 3.1, for only:

$79.95

domestic USA shipping & handling included. The package is distributed on two diskettes, and includes a reference manual, introductory primer, a set of on-line tutorials, and a support line for technical assistance.

Hardware Requirements: IBM-PC XT, AT, Compaq-Plus, or compatible equipment; DOS versions 2.0-3.1; minimum 128k, some programs require 256K.

Registered FILEMASTER owners may upgrade to FILEMAX with full credit plus 50% of price increase.

DEMO disk available for $9.95.

FILEMAX is available only from Grand Max Software Corporation. To order with VISA or Mastercard, dial:

1-800-624-3285

In Oregon: (503) 635-8295

Grand Max™
SOFTWARE CORPORATION

68 LEONARD STREET • LAKE OSWEGO, OR 97034

However, with appropriate DVP files, some of these programs may run in a window. Programs that assume they will execute in a fixed memory location or that they can access all of system memory do not run under DESQview.

DESQview coexists with programs that intercept and/or configure I/O ports, such as serial ports. These run in background mode and cannot be swapped to disk. DESQview does not change their port configurations.

Several additional features provided are an alarm clock (for an additional $5 sent with the user registration), an autodialer communications program, and a keyboard macro function. The keyboard macro capability, called Learn, more advanced than APX Core Executive's macro facility, supports pausing for fixed or variable length user input or for a specified period of time. Learn can define macros for specific applications (Program Script) as well as for DESQview (Global Scripts). Script files do not work when the applications program directly tests for keyboard input or checks the BIOS input buffer for keystrokes, instead of calling DOS to solicit a keystroke.

DESQview is a rich concurrent environment that rivals TopView. Although no developer's toolkit is available for writing DESQview applications, the documentation implies that Quarterdeck Office Systems is willing to work with outside developers who wish to make their software compatible with DESQview; a developer also may write applications that are TopView Aware, which equates to being DESQview-Aware.

DoubleDOS. SoftLogic Solutions' DoubleDOS version 3.0R runs two tasks; the limited number of concurrent processes supported leads to a simple configuration and easy use. It requires a system with 256KB to run two applications; smaller applications run with 192KB.

Start-up permits definition of a DoubleDOS CONFIG.SYS file (DDCONFIG.SYS) that specifies the display type, memory allocation, status menu type (long or short version), and menu key specification (to eliminate conflicts with applications programs). Memory allocation is as easy as TOP SIZE=HALF, which divides remaining non-DOS/DoubleDOS memory between the two tasks, or BOTTOM SIZE=192.

DoubleDOS does not support windowing, but does define VISIBLE and INVISIBLE tasks. The VISIBLE task retains control of the keyboard and display; INVISIBLE tasks take input from a disk file. Otherwise, the user would have to switch the INVISIBLE task to VISIBLE, enter the keyboard information, and switch again. INVISIBLE tasks may be suspended for optimal performance of the VISIBLE task. Cutting and pasting betwen tasks is not supported.

One nice feature, unusual for concurrent programs, is DoubleDOS's support of two displays. When the system is configured with both monochrome and color displays, one task can write to the monochrome and the other to the color display. Two-display operation eliminates problems that a concurrent environment may have with ill-behaved applications programs.

With single display systems DoubleDOS provides patch files for specific versions of popular programs to eliminate direct video I/O (direct video access causes difficulty when a task is INVISIBLE). When patching applications programs is not preferable, an ill-behaved INVISIBLE program can be suspended until the user is ready to switch the program to VISIBLE.

DoubleDOS supports automatic memory sizing for each task, batch files, and autoloading of applications files. It provides a version of ANSI.SYS that permits keyboard remapping by each task.

MultiLink Advanced is a formidable system that puts a tremendous burden on the user to fix programs that are ill-behaved. This package requires a lot of work.

It supports an 80-character type-ahead buffer for each task.

The DOS PRINT function can be run before or after DoubleDOS is executed. In general, DoubleDOS does not work with programs that terminate and stay resident prior to the start of DoubleDOS; these programs can be run as visible and invisible tasks.

E-Z-DOS-IT. This is a "many-copies-of-DOS" system. E-Z-DOS-IT requires a minimum of 256KB RAM, although more memory enables additional concurrent programs to execute.

The root of Hammer Technologies' E-Z-DOS-IT system is four set-up screens, each configuring eight partitions. Configuration consists of setting memory requirements, process type, monitor type, color set (two medium-resolution palettes), indicating whether the program is a communications file (and should receive more processing time in order not to miss characters), and indicating that an autofile exists, (and a batch file must be executed when the program is loaded).

E-Z-DOS-IT has three different process types: SUSPEND-type programs write directly to screen memory and may not run in the background; HOLD-type programs run in the background, but pause to be switched to the foreground for keyboard input; DISCARD-type programs write display output to the bit bucket until switched to foreground mode, when the most recent display is seen. E-Z-DOS-IT has no windowing or partial screen display, so background tasks are invisible.

Tests showed that two SUSPEND-type programs with heavy screen I/O are difficult to switch; random characters show up on the display until the new background task is suspended.

On-line help is available, but rarely needed; this package configures COMMAND.COM with varying memory, monitor, color, and batch file support.

The version tested was copy protected, but Hammer has announced a new version that is not copy protected.

MultiLink Advanced. At $495, MultiLink Advanced 3.02 is more expensive than the other packages reviewed. Technical complexity makes this a package not recommended for the novice.

Developed by The Software Link, Inc., MultiLink Advanced has some unique features. Up to 4.3MB of memory is available for tasks through a customized, enhanced memory specification using Tall Tree's JRAM-2 and JRAM AT boards. Background tasks can be allocated a maximum of 448KB, and the foreground task can use a maximum of 640KB. In addition, the system supports up to eight CRT terminals attached to a PC through serial ports for nongraphic, noncolor, nonsound use.

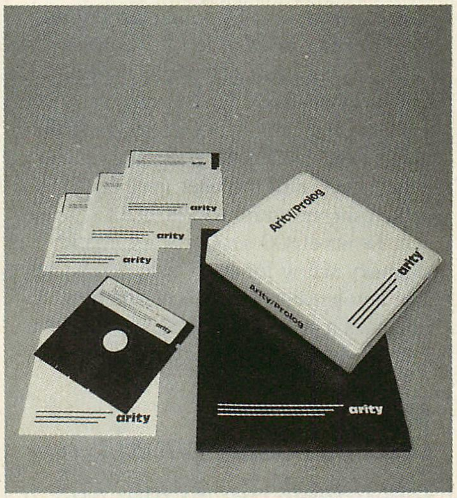
MultiLink Advanced provides several tools to coerce into coexistence those ill-behaved programs that write directly to the screen. MLVIDRAM allocates a partition area for use as a secondary video buffer area. MLVIDFIX patches the program to write its display output to the secondary video area. The documentation lists about 20 programs certified to be compatible with MLVIDFIX's patching ability. The package also includes a program called Sentry that monitors the execution of an offending program to prevent direct video I/O.

MultiLink Advanced comes complete with MLBBS, a bulletin board sys-

More Power Than You Thought Possible

Arity offers the first serious implementation of Prolog for IBM personal computers. Arity/Prolog is a powerful, highly optimized, and extended version of the logic programming language Prolog. Imagine building software applications with a language that solves problems through deduction and logical inference. The task of creating complex programs is much faster and easier, resulting in lower development costs. Arity/Prolog is now in use in a wide range of applications in industry, business, research, and education. The solution—the *Arity/Prolog Interpreter:*

- Source level debugger
- Virtual databases, each with a workspace of 16 megabytes
- Floating-point arithmetic
- String support for efficient text handling



- Interface to assembly language and 'C'
- Text screen manipulation
- Integrated programming shell to MSDOS
- Comprehensive set of evaluable predicates
- Definite clause grammar support

Arity/Prolog Interpreter $495.00

Arity also offers the *Arity/Prolog Compiler and Interpreter,* a sophisticated development environment for building AI applications. Essential for producing fast, serious production code.

Arity/Prolog Compiler and Interpreter $1950.00

The *Arity/Prolog Demo Disk* is available for $19.95. ■ Arity/Prolog products run on the IBM PC, XT, AT, and all IBM compatibles. ■ To order, call (617) 371-2422 or use the order form below.

arity corporation 358 Baker Avenue, Concord, MA 01742

Name

Organization

Address

Quantity	Product	Unit Price	Total Price
	Arity/Prolog Compiler & Interpreter	$1950.00	
	Arity/Prolog Interpreter	$ 495.00	
	Arity/Prolog Demo Disk	$ 19.95	
		Subtotal	
		MA residents add 5% sales tax	
		Total Amount	

☐ Enclosed is a check or money order to Arity Corporation

☐ Please bill my ☐ Mastercard ☐ Visa ☐ American Express

Account #

Valid ___/___ to ___/___ signature

☐ Please send me more information about Arity and Arity/Prolog

arity 358 Baker Avenue, Concord, MA 01742

M-AD-04

CIRCLE NO. 136 ON READER SERVICE CARD

tem that provides full statistical reporting of calls, time limits, message scanning, file upload/download (including XMODEM support), help, and remote access. Multiline BBS support was not tested for this review; however, the multiterminal operation of MultiLink Advanced did not perform well, which is a bad sign. A keyboard macro feature (MLKEY) is also included.

Another advanced feature is a dedicated spooler/print utility. Each partition supports a copy of the spooler program that updates a common disk file with print requests giving priority and disposition (print and delete, or hold.) The print utility prints the spooled files and provides operator ability to select or change the order that files are printed, purge files, stop/start the printer, print a portion of a file, or delete a file.

In addition to these enhancement programs, MultiLink Advanced includes MLUTIL, a general utility program to assign/deassign a serial port to a single partition, set terminal baud rates and other communications parameters, and define new CRT terminal types. The DIS function disables tasks that appear to be in a keyboard wait loop or waiting for communications. Once disabled, MultiLink Advanced awakens the task when a keystroke is entered, a character is received at the serial port reserved by the task, or every five seconds.

MLUTIL tunes system performance by reallocating time between tasks according to user-assigned priorities. System performance also is tuned by enabling/disabling the use of time slicing as a task-switching algorithm; when disabled, task switching is done only when system calls are executed.

MultiLink Advanced is a formidable system that puts a tremendous burden on the user to fix programs that are ill-behaved, as opposed to running them on a less functional level. This package requires a lot of work and technical understanding to use. It is also copy protected, a definite disadvantage.

The eight concurrent environment packages reviewed here offer varying functionality and enhancements to the DOS user. APX Core Executive offers concurrency, the DOS command line interface, and little else; however, a growing list of software companies are incorporating Core Executive into their own products when concurrent functionality is desired.

Concurrent PC-DOS is the only environment package reviewed that can replace DOS altogether, providing a complete operating system in addition to concurrency and windowing. However, C/PC-DOS is not totally compatible with DOS. DoubleDOS is the simplest of the concurrent environments; it enables only two tasks to execute simultaneously; the simplicity of this package reduces the learning time and intimidation experienced by a new user.

E-Z-DOS-IT is another simple concurrent environment that provides a DOS command line interface, while MultiLink Advanced provides concurrency and a complex set of tools to force concurrent execution of ill-behaved programs. Although MultiLink Advanced additionally provides multiuser capability and a bulletin board system, a high price tag warns users to evaluate their need for MultiLink Advanced before buying.

The best package reviewed is Windows, offering many enhancements over single-tasking DOS.

The packages offering the richest sets of features and the best attempts at an improved user interface are DESQview, TopView, and Windows. DESQview and TopView offer similar functions. DESQview, however, provides disk swapping, graphics, graphic scaling, and support of DOS 3.x function calls; TopView does not.

TopView, on the other hand, is a credible attempt to establish an industry standard for concurrent environments, and it is especially appropriate for text-based environments—computers configured with monochrome adapters and monitors. Many programs are already TopView-compatible, and some are being written to be TopView Aware. In addition, IBM can ensure that TopView remains compatible with future plans for the IBM PC, an advantage that other vendors may not have.

However, several product deficiencies must be remedied quickly. TopView must evolve to include features such as extended memory support, disk swapping, and graphics support to remain a viable environment product.

The best package reviewed is Windows, offering many enhancements over single-tasking DOS. Enhanced memory management, an excellent user interface, data interchange capability, and acceptable performance have been successfully implemented. Future evolution of Windows should also be consistent with future IBM PC announcements. Windows is clearly a product to enhance the PC user's productivity both now and in the future.

APX Core Executive: $95.00
Application Executive Corporation
600 Broadway, Suite 4C
New York, NY 10012
212/226-6347
CIRCLE 343 ON READER SERVICE CARD

Concurrent PC-DOS: $295.00
Digital Research, Inc.
60 Garden Court
Monterey, CA 93942
408/649-3896
CIRCLE 344 ON READER SERVICE CARD

DESQview: $99.95
Quarterdeck Office Systems
150 Pico Blvd.
Santa Monica, CA 90405
800/845-6621
213/392-9851 (in California)
CIRCLE 345 ON READER SERVICE CARD

DoubleDOS: $99.00
Soft Logic Solutions, Inc.
530 Chestnut Street
Manchester, NH 03101
603/627-9900
CIRCLE 346 ON READER SERVICE CARD

E-Z-DOS-IT: $59.95
Hammer Computer Systems, Inc.
700 Larkspur Landing Circle, Suite 285
Larkspur, CA 94939
415/461-7633
CIRCLE 347 ON READER SERVICE CARD

MultiLink Advanced: $495.00
The Software Link, Inc.
8601 Dunwoody Place, NE, Suite 632
Atlanta, GA 30338
404/998-0700
CIRCLE 348 ON READER SERVICE CARD

TopView: $149.00
IBM
Entry Systems Division
P.O. Box 1328
Boca Raton, FL 33132
CIRCLE 349 ON READER SERVICE CARD

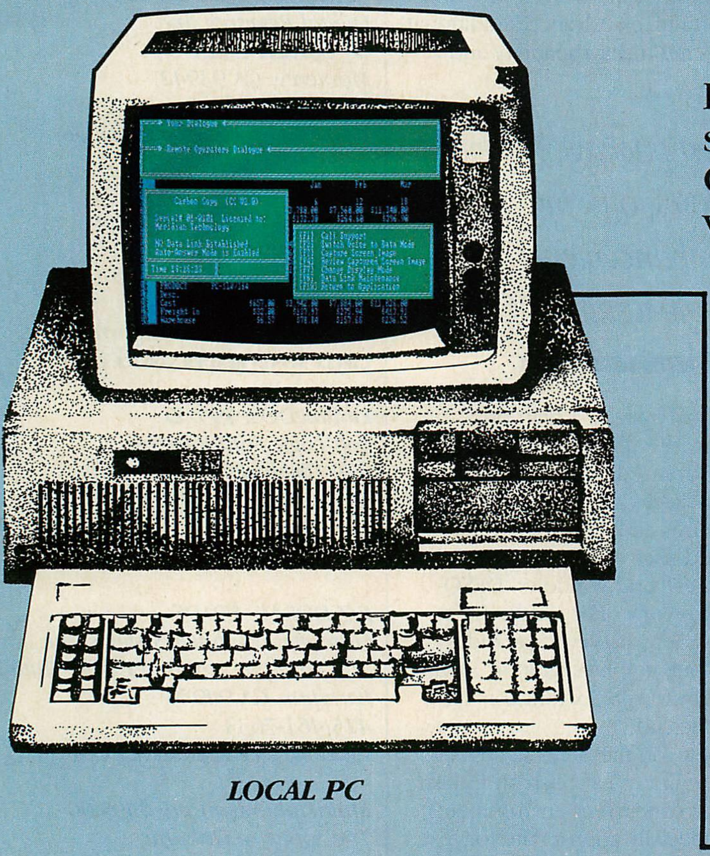
Windows: $99.00
Microsoft Corporation
10700 Northrup Way
Bellevue, WA 98009
800/426-9400
206/828-8080
CIRCLE 350 ON READER SERVICE CARD

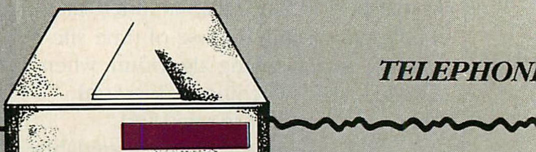
Don Awalt is director of corporate information services for EMC Controls. He is the author of Hellcat Ace, by MicroProse Software.

Introducing

The ultimate

Finally, an acceptable way to run personal computers remotely. Carbon Copy is a powerful software tool which allows you to instantly connect your PC with any other PC. Once connected, run any program you wish. Application programs, hardware or software diagnostics, utilities, or any other PC program of your choice can be used remotely with Carbon Copy.





- Carbon Copy™ has many uses. Originally, it was developed for use in major corporation support departments, where it has saved thousands of hours in needless telephone calls. Now, the support personnel can *SEE* what the problems are, *INSTANTLY.* No more wasting valuable, productive time in unnecessary endless conversations with an inexperienced user trying to describe a highly technical problem.

- Many major corporations use Carbon Copy as a training tool. They simply call up their remote user (a client, a customer, or perhaps a branch office) and proceed to demonstrate how to use the custom software package developed in-house by their programming staff. The remote user is able to ask and answer questions, working with the training staff in real time, without any staff having to leave their own office.

- Corporate sales departments use Carbon Copy with their portable lap top computers to enter orders into the main PC back at their office. They can even check the current sales projections in their Lotus spreadsheet, as well as pick up their calls and messages from any telephone on the road.

THE DEAL:
$6,995
To help you start your network system, ALR has packaged a powerful AT Tower, with two turbo XT workstations all networked and ready to perform.

IBM is a registered trademark of IBM Corporation.
Xenix is a registered trademark of Microsoft Corporation.

AT—THE ALR SYSTEM 286

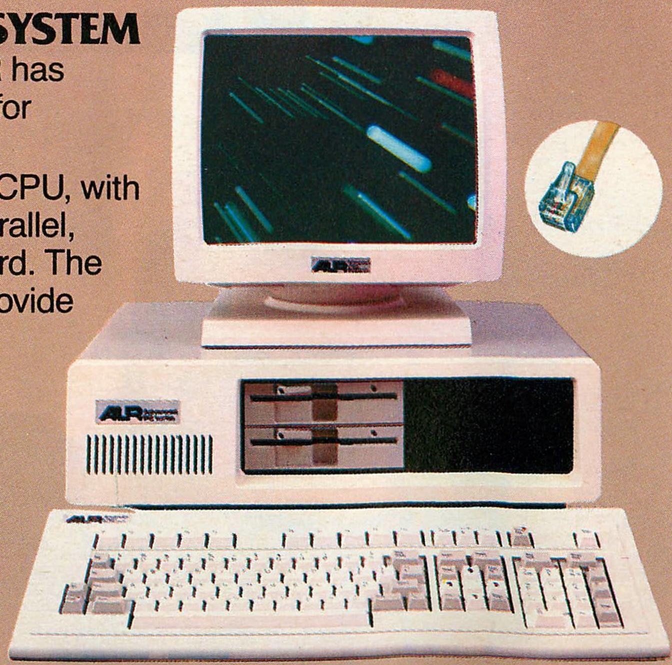
The ALR System 286 is based on the advanced 80286-8 16 bit microprocessor with a system clock rate of 8MHz. This performance results in running most IBM PC or XT software up to 350% faster: In seconds you'll be able to recalculate large spreadsheets and instantly load files.

COMPATIBILITY BY DESIGN. At Advanced Logic Research, the System 286 was designed to be 100% IBM PC AT bus compatible with full attention to the BIOS ROMs resulting in total support of IBM DOS 3.0, 3.1 and Xenix operating systems.

XT—ALR'S MOST POPULAR SYSTEM

On the Advanced XT Turbo System, ALR has installed all the features leaving little need for add-ons.

The Advanced XT System has a 65% faster CPU, with 640K RAM and a 20 megabyte hard disk. Parallel, serial and clock calendar ports are all standard. The swivel and tilt monitor and video controller provide sharp characters plus 720 × 350 graphics.



For further information on your ALR system, call
(714) 666-2951.

CIRCLE NO. 116 ON READER SERVICE CARD

ALR

ADVANCED LOGIC RESEARCH, INC.
2991 E. WHITE STAR AVE.
ANAHEIM, CA 92806
(714) 666-2951

The Futility of Copy Protection

A discussion of copy-protection techniques reveals how ineffective they can be.

DAVID SMALL

Copy protection is the subject of fierce debate in the software industry. Many of the arguments on both sides of the issue, however, are based on misinformation. Even the term *copy protection,* for example, is a misnomer; all diskettes can be copied.

A diskette must be readable in order to be usable; the information on it must be transferrable to computer memory. Hence, whatever is on the source disk can be read. In addition, all computers can write whatever is in their memory out to a destination disk. The more appropriate term for preventing the unauthorized duplication of a working computer program, then, is *execute protection*. Any program can be copied, but with certain protection measures, it may not be executable after it is copied. This is probably the most important concept of copy protection, and also the least understood.

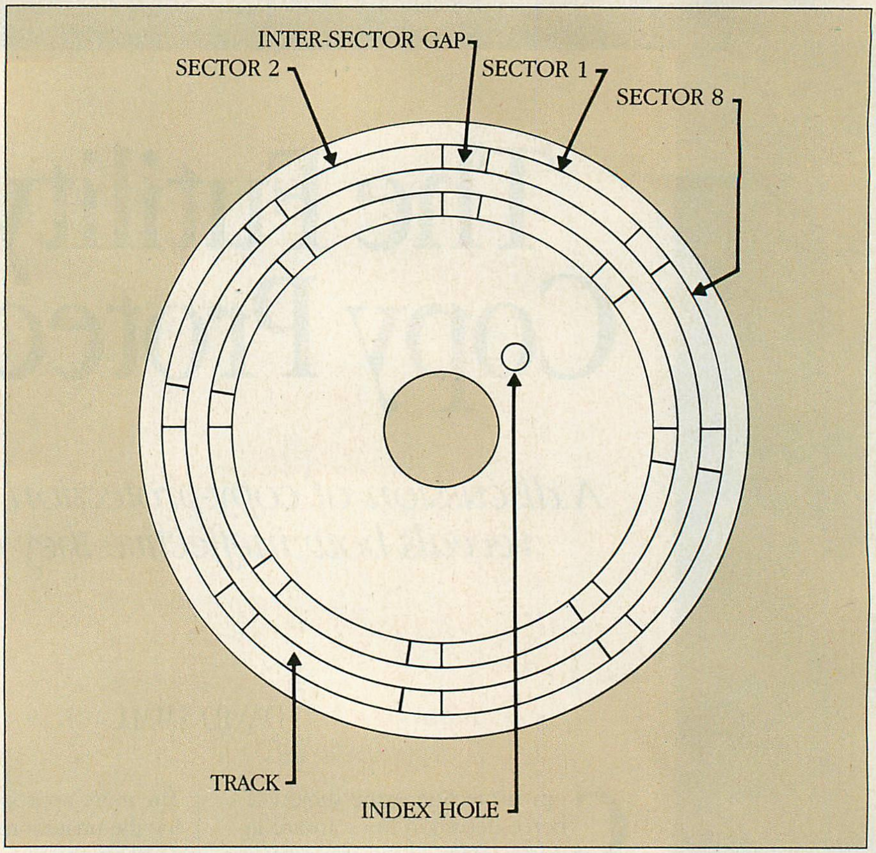
The basic design philosophy of most copy-protection schemes in use today is that the source diskette must be uniquely defined. When the program starts, it checks the diskette from which

ILLUSTRATION • CHRISTOPHER ZACHAROW

it was loaded. If the diskette has the unique mark, the program proceeds; if not, the program terminates. The mark *must have* three attributes: it must be unique, uncopyable, and detectable. Copy protection companies use different terms for this mark—*signature, fingerprint, unique identification.*

The effect to the end user is that while the program is the same from either the original or copied disk, it will not execute if loaded from a copy. So, the disk has not really been copy-protected, but execute-protected.

FIGURE 1: *Typical Diskette Layout*



The tracks of a typical floppy diskette are concentric circles that are numbered successively from the outermost track to the center of the diskette. Each track is divided into sectors that are further separated by inter-sector gaps.

FLOPPY-DISK GROUNDWORK

Floppy diskettes are circular surfaces of magnetizable material that are rotated by a disk drive at 300 RPM, or five rotations per second. They have 40 tracks in concentric circles; data occupy these tracks, which are divided into pie-shaped sectors. IBM uses eight sectors for DOS 1.0, and either eight or nine for DOS 2.0 and 3.0 (see figure 1).

To read or write a given sector, the read/write head of the disk drive is moved in or out to the proper track. The system waits for the proper sector to rotate under the head, then reads or writes data from/to the disk. All normal data transfers to the disk are done in sector-sized increments; reading or writing individual bytes is not done. The operating system converts requests for non-multiples of sector-sized data into sector-sized requests.

At the beginning of each track is an index pulse. Each sector on a typical track has two parts: a sector ID header and the actual sector data. Common terms for this header are *sector ID mark, ID mark,* and *ID address mark* (IDAM). The mark is six bytes on the diskette, with special header and trailer bytes that identify it to the floppy controller IC as a sector mark and not as data. The six bytes are, in order:

1. Track number or cylinder number, 0-39 inclusive.
2. Side number or head number. Front side of the diskette is 0, back side is 1; single sided drives are 0.
3. Sector number or record number (this is the number of sectors per track), 1-8 or 9, inclusive.
4. Sector size or number of bytes, 0, 1, 2, or 3— the codes for 128-, 258-, 512- or 1,024-byte sectors, respectively. The default size for a diskette formatted under DOS is 512.
5. and 6. A CRC to ensure the above information has not been altered and is valid, not just accidental bits.

The four useful bytes of the sector ID mark (numbers 1 through 4 above) are referred to as CHRN. The CRC bytes are a 16-bit checksum of the CHRN and are calculated when the sector is written. When the sector is read, the CRC is recalculated based on the data read in the CHRN and compared to the data read in the CRC bytes. Any discrepancies are returned as a CRC error.

Following the ID header are a few scratch bytes, a beginning of data (BOD) mark, and 512 bytes of sector data. Two bytes of CRC follow the sector data. Again, this is a checksum of the sector's data, calculated when the sector is written. Discrepancies between the written and the calculated CRCs are returned as a CRC error and a data error. After the first sector's CRC is the inter-sector space, then the other sectors and inter-sector spaces.

All disk I/O is done via the floppy-disk controller IC. Therefore, checks for a copy-protection mark must go through that IC. Any attempt to read a sector that is not present will cause a "Record not found" (RNF) error to occur after five revolutions.

From a programmer's standpoint, the diskette can be written to or read from in several ways, a critical point to remember in copy protection. First, the operating system can be used, which makes sector sizes invisible to the programmer and allows the operating system to determine the sector allocation. An example is DOS COPY.

Alternatively, the programmer can make calls to the BIOS to read or write a specified sector. This is done by using the BIOS interrupts or by calling the ROM routines directly. However, the BIOS interrupt method, which is the most commonly used, leaves the program vulnerable to being copied.

Third, the floppy controller IC can be commanded directly with IN or OUT machine language statements, and custom code can be written to transfer bytes, check CRC, and so forth.

SECTOR COPYING

Several copy-protection techniques are designed to prevent sector copying of a diskette. The sector copier reads each sector of each track, disregarding DOS's tables for which sectors are used and which are not, and writes them back out. It copies an empty disk the same

way as a full disk and does not care what information is in the sectors.

If the standard format of the floppy disk is changed, the disk will have a unique characteristic. If the change is detectable, the diskette is marked. As sector copying programs become more sophisticated, however, some of these format changes are rendered less effective as protective measures.

One such technique is to cause a known RNF error on the original diskette by omitting a sector. When the copy-protected program makes a read request to the sector that is missing, either through the BIOS or direct calls, the floppy controller should return an RNF error. If an RNF error comes back, the program continues; if not, the diskette is not original.

An often used variation on the RNF theme is deliberately to generate a CRC error on a given sector. A standard technique is to issue a write request for that sector, start transferring data, then remove the disk drive motor on signal. This garbles the data in midsector and ensures that the CRC bytes at the end of the sector are not updated; therefore, every time that sector is read, a CRC error will be returned.

A problem with RNF and CRC errors is that part of the diskette is unavailable to the operating system; the sector is either bad or not there at all. If a user tries to use that sector for legitimate reasons (data storage, for example), a disk error will occur.

An alternative copy-protection scheme is to alter the sector interleave. This process does not render any of the diskette unusable and remains invisible to the user. Instead of a sector order of 1,2,3,4,5,6,7,8,9, this scheme might use 1,6,2,3,8,4,9,5. The sectors will still be on the diskette, but out of order. The time between sectors can be measured and the differences in retrieval times between the standard sector order and the modified order can be detected.

The sectors also could be numbered according to a nonstandard system. Several sectors on a particular track could be numbered the same—for example, 1,2,2,2,2,2,2,2,9. The floppy controller will not notice anything out of the ordinary; when a request of sector 2 is made, the first sector 2 that is found will be returned.

On the special track, different data are written in each sector 2. In the program that is being protected, sector 2 is read several times in succession and the data returned are checked for differences at each read; a different physical sector is being read each time.

Another option is to change the length of the sectors. Eighteen 256-byte sectors could be put on one track, instead of nine 512-byte sectors. That special track can be checked in several ways: by reading sector 18 or by checking that the track has 18 sectors or that the sector lengths are 256 bytes.

Sector numbers with very high values are often used in place of numbers 1-9. During the boot process, a read is done to the high sector number. If an RNF occurs, the program aborts; if, on the other hand, the high sector number is found, the program continues.

NEWER TECHNIQUES

None of the sector-copying techniques discussed above is very effective today, because newer, sector-oriented copy programs are able to duplicate format changes. The latest generation of copy-protected programs, therefore, must use techniques that provide proof against even these sector-oriented copy programs. In order to penetrate the newer protection schemes, a copy program must be sophisticated enough to measure a large number of inter-sector gaps, duplicate them correctly, and preserve the track-to-track synchronization.

One of the newer methods is to change the rotation speed of the disk. A standard disk drive rotates at 300 RPM.

As sector-copying programs become more sophisticated, some protection techniques become less effective.

If that speed is reduced to, say, 280 RPM, the data are written at the same rate, but due to the slower speed the data density on the diskette is greater and each track has room for an additional sector. This is a reasonably secure system because ten sectors can fit onto the track only if the disk drive speed is modified. The index pulse will occur during the middle of the last sector if the data are written with the disk drive rotating at 300 RPM.

This approach has a few problems. The disk drives must be modified to write the track, which prevents use of the popular mass duplicating services. More importantly, altering the basic operating parameters of the disk drive could reduce the read reliability. Writing ten sectors of data onto a single track means that the data are read back at a faster rate due to the increase in data density. Floppy-disk drive controller manufacturers assume the disk is rotating at 300 RPM during the design phase. Changing the data rate by increasing the data density erodes the tolerances available for the floppy disk drive controller to synchronize with the disk rotation to read the data. Some disk drives now in use will reject a disk with an extra sector, because the synchronization bytes that occur at the start of each sector are not the standard length. These have on-board separators that reject the data before they reach the floppy drive controller. Use of this type of copy protection by software manufacturers could cause serious problems with reading original disks.

A recent development in sector-oriented copy programs is their ability to use software to reduce the speed of the disk drive. The motor-on signal to the disk drive is oscillated at high speed (reducing the duty cycle to less than 100 percent), which reduces the speed of the disk drive. For example, a 93-percent duty cycle will result in the drive rotating at 280 RPM. If the copy program does this cycling as it is writing the destination track, it will copy all ten sectors correctly, because the track will have room for ten sectors.

Another copy-protection scheme involves writing the data on one track with a known spatial relationship to the next track. If the second track has several sectors with the same number containing different data, the program can read the first track and then read the second track immediately. If the correct version of the sector is read, the disk is original; however, if the spatial relationship between the tracks is not maintained exactly by a copy program with respect to the index pulse, the program is able to detect the discrepancy.

The inter-sector distance also can be adjusted. This does not have a rigid specification defined by the floppy disk controller. Small variations in the time between reading two sectors can be made by altering these lengths when the diskette is formatted. The program then can check for discrepancies.

Programs can also be protected by putting nonstandard codes on the diskette track, using either special hardware or a Western Digital controller, which the NEC controller used in the PC cannot duplicate. The codes then are read and examined by the program.

These are fairly effective methods of copy protection at this time, although they are not airtight. The protection

scheme must work on a variety of equipment and the tolerances cannot be too tight or failures will occur even when using original diskettes.

OUTSIDE THE LIMITS

Writing data onto a track outside the normal limits is another form of copy protection, although it reduces read reliability. A diskette has room for additional tracks (40 through 42), which the floppy-disk controller can access. Some drive mechanisms will not go past track 39, however, and others may be damaged if they do. This form of copy protection is not considered viable.

A fairly effective method, sometimes referred to as the *fuzzy sector technique*, is to write data of indeterminate strength on the diskette. The data could be read from the diskette as either a 1 or a 0. Repeating the read cycle several times would not return the same data each time. Some diskettes may encode the data as normal, however, because of the tolerances in the magnetic media, and some disk drives would read the indeterminate data as the same on each attempt. These original diskettes may be considered copies even though they actually are originals. A few copy programs now have the ability to duplicate this feature under program control by selecting and deselecting the disk drive at very high speed during the write process.

The mass duplicating houses are not willing to use this technique because of the hardware modifications needed to generate soft bits (generally, some sort of reduction of the current through the read/write head).

One of the better known copy-protection techniques is physically to alter a small area of the disk so that it is no longer able to hold a magnetic field. This is a unique, uncopyable, and identifiable characteristic. The Vault Corporation's Prolok disks use this method. Track 39, on the front side, has a damaged area on it.

The first Prolok disks simply read the sector in the bad area and determined that it had a CRC problem. Unfortunately, this CRC error was readily copyable; hence, many Prolok disks were copied easily. Newer Prolok versions, however, are more sophisticated; they write to the affected area, then check to make sure the write failed because the media are damaged. This obviously is not readily copyable. Unfortunately, this technique requires that the diskette not be write-protected during use. The scheme is also questionable because deviations in the surface of a diskette are bound to be detrimental. Prolok's manufacturer claims that drives are not affected.

These Prolok diskettes are expensive, because each one must be individually burned. A software house can buy standard diskettes for less than $1 each, while a Prolok sample pack includes three diskettes for $15. Although the Prolok technique is perhaps the most effective copy-protection method available, its drawbacks make it a questionable choice. Furthermore, certain software techniques (discussed below) can bypass Prolok.

HARD-DISK ANNOYANCE

One extremely annoying aspect of diskette-based copy-protection techniques is that they do not work with hard disks. When a copy-protected program is moved to a hard disk, nothing tells the program that it is no longer running from a floppy disk. Therefore, when the copy-protection check is run, the floppy drive is switched on by the program, and the usual tests of the floppy disk are made. The copy-protection check usually addresses the floppy controller IC directly, so it bypasses the hard disk completely. The effect to the user is that the original floppy must be present in the disk drive in order to bypass the protection check.

Programs can also be protected by putting nonstandard codes on the diskette track, using either special hardware or a Western Digital controller.

A recent scheme, aptly described as *install-deinstall*, supports hard disks. When installed on a hard disk, the copy-protected software marks the hard disk in some manner. Also, the floppy disk is deactivated, so it cannot be copied to another hard disk. If the customer wishes to move to a new machine, the program must be deinstalled, which reactivates the floppy and deactivates the hard-disk copy.

A hard disk can be uniquely identified in several ways. The first is to check through the bad track information that the factory puts on a hard disk during the testing/run-in process. Most hard disks seem to have one or two bad tracks, and this information is contained in a table that the controller consults to avoid bad tracks. The bad tracks differ from disk to disk.

A sector on the hard disk that is not commonly used (such as a sector in the boot area or near the end of the diskette that the manufacturer did not use) also can be marked/demarked. These schemes have two problems from the software manufacturer's viewpoint. First, they are hardware-dependent. The formatting positions for unused tracks on available hard disks may not be standard. Second, the program must go through the BIOS to use a hard disk if aftermarket drives are to be supported. This leaves the program vulnerable to programs that specifically allow a floppy protected program to run off a hard disk by trapping and modifying calls through the BIOS.

The addition of any data that are inaccessible to the owner of the hardware is a questionable practice. Any scheme that examines the bad sector data as a fingerprint for the program to run also must accommodate sectors that go bad on a hard disk during use—without reinstallation being necessary.

A move to standardize copy-protection devices is being promoted by ADAPSO, an organization of software manufacturers dedicated to fighting disk piracy. After studying disk copy-protection schemes at great length, ADAPSO concluded that current methods are ineffective against copy programs and expensive for software manufacturers. ADAPSO suggests using hardware *keys* that plug into the serial port of an IBM computer; software manufacturers can interrogate a key to check for validity, yet normal operation of the serial port still is permitted. Each key would be numbered; a key for one software package would not work with another. This approach is being backed by companies such as Lotus and Microsoft.

ADAPSO's plan, however, may be doomed before it ever gets implemented. First, it will have to avoid the BIOS and talk directly to the UART IC on the serial card. Supporting all the different serial cards available would be difficult, if not impossible, so the user would be restricted to a few popular cards. Second, a hardware "demon" (see discussion below) to watch the serial bus and duplicate the actions of such a key would be easy to make with a small processor of the inexpensive 8048 variety and an interface (1488/1489 IC). Such a hardware demon could replace all the keys for different packages.

Third, the ADAPSO keys would raise the cost of a software package considerably. These keys are going to cost the manufacturer at least $10, according to ADAPSO. Fourth, the question remains of how the key will interact with software that uses the RS-232 port. Fifth, the key adds another layer of support problems to software manufacturers. Finally, many users will find ways to bypass the keys as soon as they are made public.

COPY TOOLS

If a diskette is copy-protected with one of the more sophisticated techniques using unique marks, the program on the disk will be copied, but that program may not work on the copy. The diskette has been copied; the unique characteristic has not. Manufacturers of copy programs get around this obstacle by designing their programs to examine the code on the diskette, looking for well-known programs (Lotus 1-2-3 would be a good example). If the code is recognized, a special patch is inserted into the disk program to bypass the copy protection. This leads to a race with the software manufacturer to change its code, often resulting in monthly updates of the copy program.

Beyond copy programs, a serious copier has no end of tools available—logic analyzers, tracers, and nonmaskable interrupt tools. A recent addition to copy tools is the analog flux copier, also called bit copier or transition copier. This copier reads the source disk at the bit level (the transitions of magnetic flux) and writes at that level to the destination disk. No processing or examination of the data is done; whatever is on the source disk, magnetically, is copied to the destination disk. If the source disk is copy-protected, so will be the destination disk.

Another copy tool is the analog board, introduced by Central Point Software. It has added circuitry that helps it handle soft bits and low RPM tracks, common protection techniques. The only copy-protection technique that cannot be copied with this board is the Prolok disk-damage technique. However, Central Point is developing a new board to handle the Prolok technique by aborting the write between the controller IC and the diskette when the Prolok sector is accessed.

DEMONS

An increasingly popular alternative for breaking copy-protected programs is the *demon*, the name of which derives from Maxwell's demon, a mythical creature controlling an atom-sized door suspended in midair. Maxwell's demon would allow quick-moving atoms to go through the door; he would stop atoms that moved slowly. Such a demon would thus create hot and cold areas in air and reverse entropy.

In a computer context, the demon is loaded into memory to intercept all disk requests; each request requires the demon's permission to reach the floppy disk controller. The demon works in two steps. First, it watches the original diskette load and notes how the copy-protection process works. In this mode, the demon is passive, merely building up a table of the load process.

Next, a copy is made of the diskette, and the demon helps get the copy running. If, for example, the program uses the CRC error technique on track 39 and a sector read is made to track 39, the demon will intercept the read request and return a CRC error status without asking the floppy controller IC. The program will not know it is talking to a demon instead of the controller; it will think a real CRC error happened and will continue.

The demon is loaded into memory to intercept all disk requests; each request requires the demon's permission in order to reach the floppy-disk controller.

Figure 2 shows the normal disk I/O process, with either DOS or BIOS (INT call) calling the BIOS ROM, which then issues a command to the floppy controller. The status is read by the BIOS ROM from the controller and returned to the calling program. Figure 3 shows the demon intercepting the BIOS call and returning the CRC error status.

Demons are inserted into the BIOS jump table on the PC. IBM uses the software interrupt to access the BIOS, INT 25H and 26H. This INT is vectored through a jump table to BIOS locations; that jump table, due to the 8088's architecture, is in RAM. A demon's address can be placed in the jump table instead of the BIOS address. IBM programs that call the BIOS directly, thus avoiding the INT call, bypass demons; but obviously, changes in ROM software would cause such programs to fail.

Demons are effective against only two of the three ways of accessing the disk; those going through the operating system and the BIOS. Either of these methods use the jump table, which can be modified for a demon's attention. Programs that address the floppy controller IC directly bypass the demon.

This poses a real dilemma for software developers. The floppy controller IC should not be directly addressed; IBM will not guarantee that it or the hardware will not be modified. As an example, if the floppy controller is accessed directly by the program, then the program cannot be transferred to a hard disk. Floppy disk requests should go through the BIOS, and that makes the program susceptible to demons.

Alternatively, the program can reset all of RAM to prevent a demon from coexisting with the program. However, this also prevents resident programs, such as SideKick, from coexisting with the program. Some demons are now sold as ROM IC, which are changes to the system ROMs, and this prevents copy protection of programs.

Quaid Software's CopyWrite supplies a demon, called RAMKEY, with its product. RAMKEY can be used to make copies of software that is protected by Vault's Prolok method. RAMKEY appears to confuse Prolok's interpretation of the damaged disk area by returning a CRC error on any access to that area. Vault Corporation has filed a lawsuit against Quaid Software for violating the shrink-wrap agreement included with all Prolok diskettes, which forbids disassembling the Prolok code. Vault claims that Quaid must have disassembled the Prolok diskette in order to produce RAMKEY (see "Shrink-wrapped Enforcement," Legal Brief, Max Stul Oppenheimer, September 1985, p. 177).

DISASTROUS CONSEQUENCES

One particularly chilling facet of the copy-protection issue is what happens to a program on which a copy is attempted. Most programs merely are halted. A new booby-trap trend has emerged, however, which may cause problems later. A classic example is an accounting program that, if it determines it is a copy, works fine until the end of the month book balancing, at which time the data disk is destroyed.

Another example is a "worm" released into computer memory that slowly eats away program code until the system crashes. The results are totally unpredictable, from a system freeze-up to reformatting of hard disks to possible physical damage (for example, the dis-

THE MOST POWERFUL, MOST COMPLETE dBASE PROGRAM DEVELOPMENT TOOL EVER!

FLASH CODE™

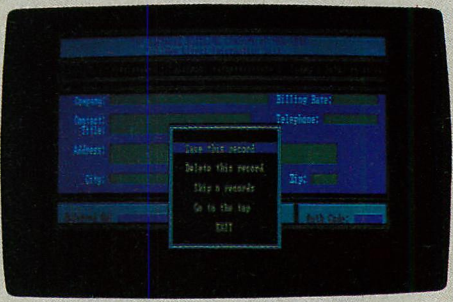
Program Developer for **dBASE II** & **dBASE III**

INTRODUCTORY PRICE $150.

Actually "DRAW" your dBASE screens!

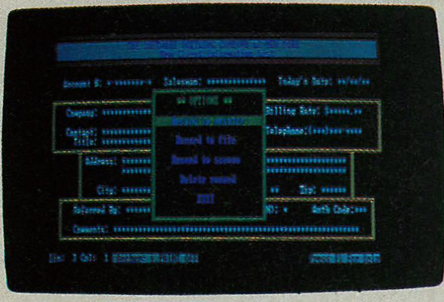
Take full advantage of all special characters, draw boxes and lines, use the color select menu, move pieces of the screen around, and more!

Specify variable names, file variables, data types, acceptable data ranges, pictures for edit checking, initial values, calculated fields, indexes, and more in a flash!



FLASH Code writes your program in either dBASE II or dBASE III!

Use it as is, modify it, or merge it with your own code. You can even use our new FLASHUP module to instantly flash up even the most complicated screen designs. No more waiting for slow print statements to display your screens.



POP-UP Windows

Create pop-up menus and pop-up help screens with our window editor. Use the FLASHUP module and have them pop up all over your dBASE applications.

NO RISK DEMO OFFER! TRY IT—YOU'LL USE IT!

Here's a no-risk offer. Order now and you'll also get a full demo disk. Use the demo and the manual for 30 days. If you don't love it, return the package for a full refund!

THE SOFTWARE BOTTLING COMPANY OF NEW YORK

Credit card orders only call 24 hours a day, 1-800-824-7888, operator 268.
For all other orders and inquiries call or write: The Software Bottling Company of New York, 6600 L.I. Expwy, Maspeth, NY 11378, (718) 458-3700. If we're shipping to a NYS address, please add 8¼% sales tax.

CIRCLE NO. 192 ON READER SERVICE CARD

FIGURE 2: *Disk I/O Process*

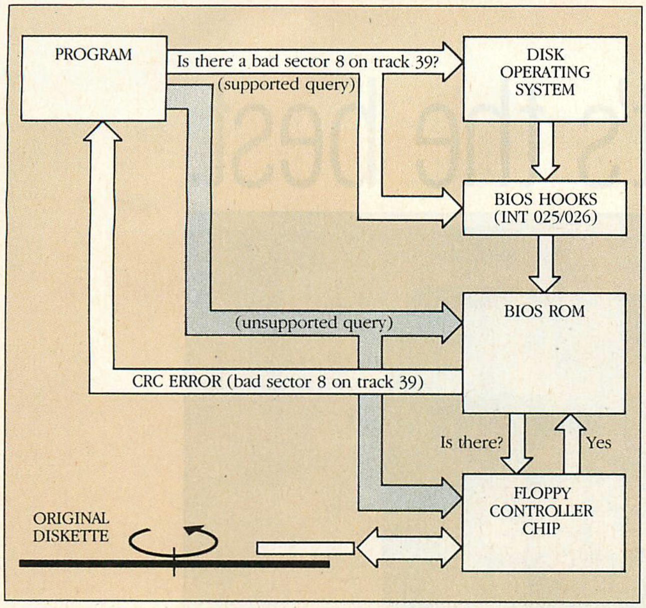
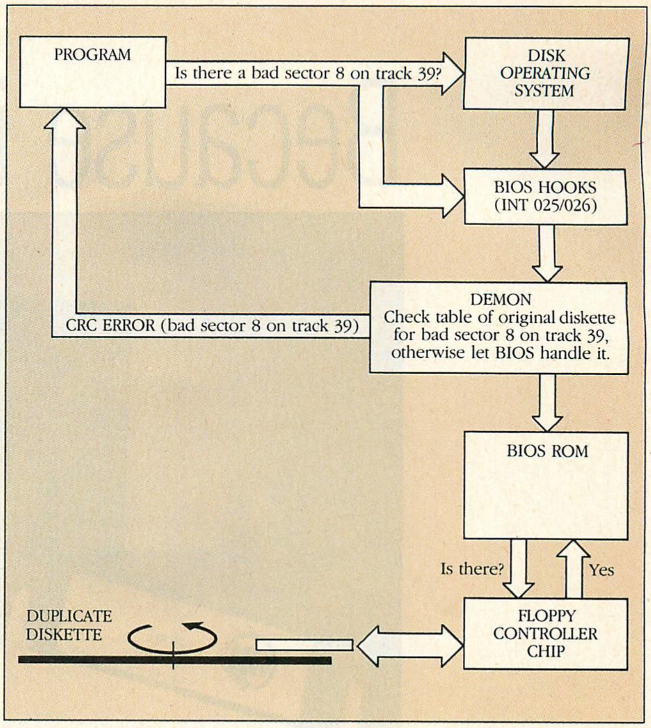


FIGURE 3: *Demon Interception of BIOS Call*



The normal flow of the program to the floppy controller for checking the format of the disk is shown in figure 2. Figure 3 shows the demon, which interrupts the flow to return the required response for the copy-protected program.

play monitor can be improperly instructed so that the video beam burns holes in the phosphor).

The general reaction to the announcement of such schemes has been outrage among users and rethinking of the process by manufacturers. An attorney recently filed a lawsuit for "willful destruction of private property" against a software manufacturer whose copy-protection scheme went berserk and reformatted one of the attorney's data disks (see "The Tort of Copy Protection," Legal Brief, Max Stul Oppenheimer, March 1985, p. 177).

Several trends are pointing to the eventual decline of so-called software piracy and a lessened need for software copy-protection schemes. One such trend is the increasing availability of easier ways to break the protection schemes. Several demons, for example, are in the public domain.

All of the copy-protection schemes discussed here share a common characteristic: they require a check in the program being protected of the original, source disk. A machine language programmer can alter the program to bypass this check. This is called *cracking* the software and is a very popular activity. Many bulletin boards specialize in *patches* or *zaps* that disable the protection schemes of popular programs. One well-known technique for "de-Proloking" software involves object code changes via IBM's DEBUG.

For many programmers, this is a game—a challenge to determine and then disable the copy-protection technique being used in a particular program. Given the current generation of debugging tools, the task is not all that difficult. A sufficiently smart copy program can copy just about any characteristic. What cannot be copied with a copy program can usually be made to work with the help of other tools.

Copy protection started in the early days of computers when software was shipped in plastic bags with one page of documentation; the cost to the manufacturer was minimal and support was virtually nonexistent. The manufacturers were in a high-risk business and wanted corresponding profits, not copies.

Today, the diskette is no longer enough; the multipage manual is necessary, along with the company's hot-line phone number, and other forms of support. The days when a program and manual could be easily duplicated are gone; now, copying the program and documentation is fairly expensive and leaves the user without support from the software house. In other words, much of the essential material (primarily support) is no longer copyable.

Software pirates still exist, but more often than not, the programs they copy are never used. The pirate may have Lotus, Asteroids, SuperCalc, and Missile Command, but if he has no instructions, the programs are *not usable.*

The trend toward lower cost software also is helping to stop unauthorized software copying. The urge to copy a $50 software package is much less than it is for a $500 package. At some point the cost of the copy program, photocopy machine, and copier's time exceeds that of the package. Borland International's Turbo Pascal and SideKick are often cited as examples of non-copy-protected products that, nonetheless, are hot sellers. Borland offers good products, with fine manuals (difficult to photocopy because they are paperback-bound) and good support.

Furthermore, copy protection adds greatly to the cost of software packages. Part of the incentive for pirating programs is their high cost; some of that high cost is the copy protection. Thus, an endless loop is created.

Another reason that copying software may be declining is the increased corporate use of computers. Except for a few publicized cases, most companies do not engage in software piracy. What they require is documentation and support from the software company.

Copy protection is an anomaly that tends to hurt harmless users and does not slow down dedicated pirates. It is a classic case of the lock on the door that keeps out honest people.

David Small is the author of three computer books. He has a computer science degree from Colorado State University.

PERFORMANCE

You would expect this kind of performance from a data base management system costing hundreds of dollars more!

GP-Tools provides you with the complex computational capabilities required to manage a variety of user-defined data types.

Handles up to fifty fields per record.

Maximum record size of 1920 bytes.

File size limited only by hardware configuration.

Instantaneous screen displays.

Data retrieval in a fraction of the time found in other systems.

Password option for each master view or subview to give maximum data retrieval and update protection.

Data encryption to prevent unauthorized peeks at the data file contents.

Up to 99 subviews of a pre-defined master view.

FLEXIBILITY

No more complex and frustrating installations or setups!

Fixed disc users need only to copy the program discs to their desired directory or subdirectory and begin executing GP-Tools.

Floppy disc users can begin executing GP-Tools right away.

No need for disc 'swap-outs' or returning to the operating system when you want to move from GP-File to GP-Print.

GP-Tools allows your master view data file to grow as your requirements grow. Redefine the master view when more information storage fields are required or when you no longer need an existing field.

USER INTERFACE

Up and running within a matter of minutes instead of months.

Stop floundering in unfamiliar territory. GP-Tools provides continuity in function key assignments and screen displays.

Each display provides a list of available actions so that you never feel lost or trapped.

Complete help facilities provided throughout . . . should you need them.

Look at your data file attributes either graphically or statistically! (No graphics card required!)

GUARANTEE

We feel as strongly as you do that a product should be of benefit to the user. It should help to increase productivity and not frustrate the user in the process.

If you are not satisfied with GP-Tools, return it to us for a complete refund.

We offer this policy because we are a no-nonsense company. We are aware that because GP-Tools is not copy-protected, it can be copied and then returned for a refund. You have placed your trust in us to bring you the price and performance that today's technology demands, and we place trust in you.

PRICE

There are no strings attached to GP-Tools. When you buy your first copy of GP-Tools, just fill out and return the registration card to us. You will be placed on our GP-Tools newsletter mailing list. The newsletter provides you with timely information on utilization and application development strategies.

You will also receive update notices as well as the updates (update charges will be listed in the update notice).

TECHNICAL SUPPORT

Allen-Harris Industries, Inc. believes that a company should stand ready to assist its customers without charging them a subscription fee for technical support.

If you have any questions about the operation of any GP-Tools product just pick up the phone and call us toll-free at 1-800-821-2165 and our technical support personnel will assist you.

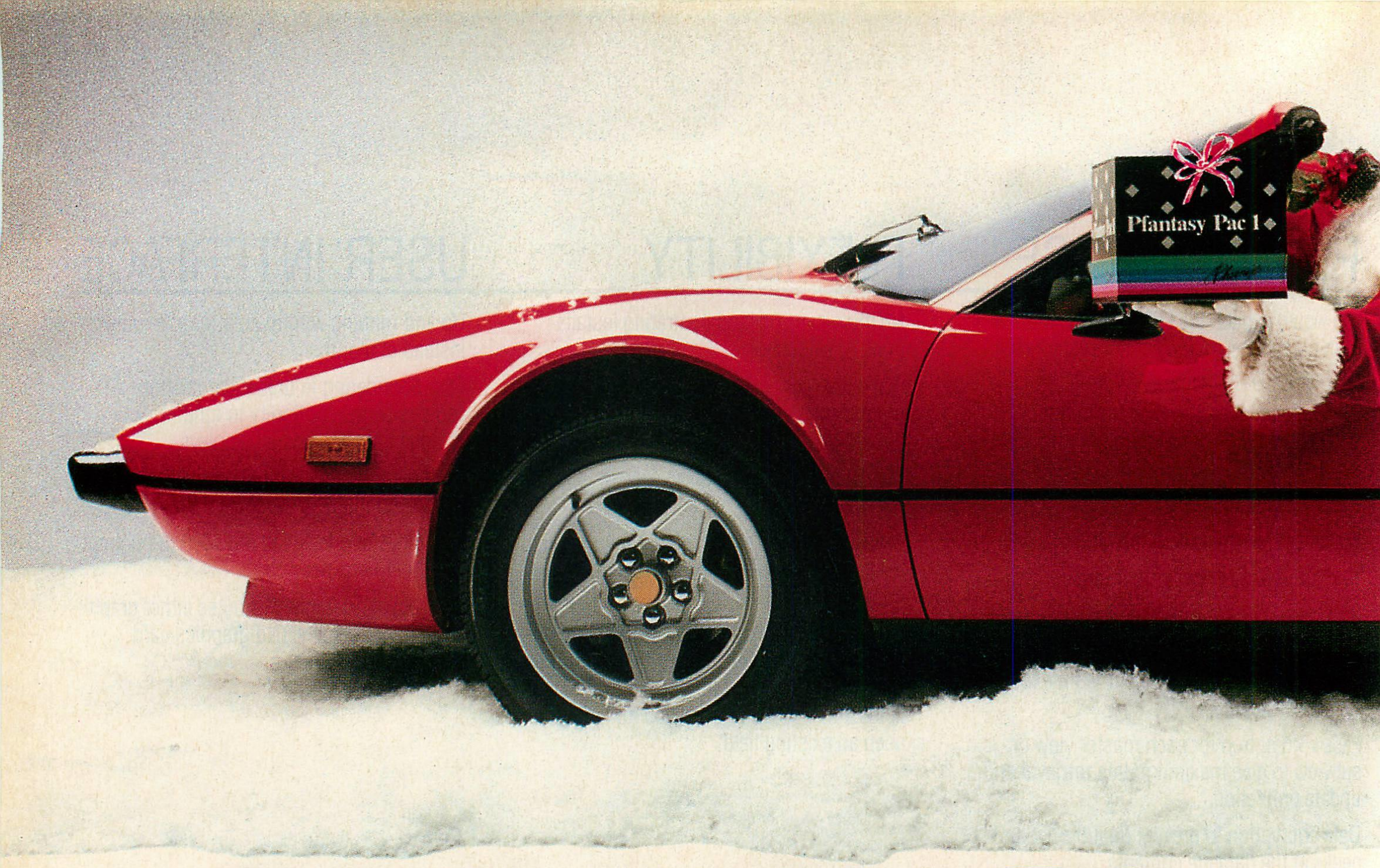
Order GP-Tools at $69.95 by calling toll-free: 1-800-821-2165
(In Florida, call collect: 305/657-1144)

Major Credit Cards Accepted

ALLEN-HARRIS INDUSTRIES, INC.
130 UNIVERSITY PARK DRIVE - SUITE 185
WINTER PARK, FLORIDA 32792

TELEPHONE 305/657-1144
CIRCLE 135 ON READER SERVICE CARD

GP-TOOLS a division of Allen-Harris Industries, Inc.
GP-TOOLS includes GP-File, GP-Print and GP-Utility.

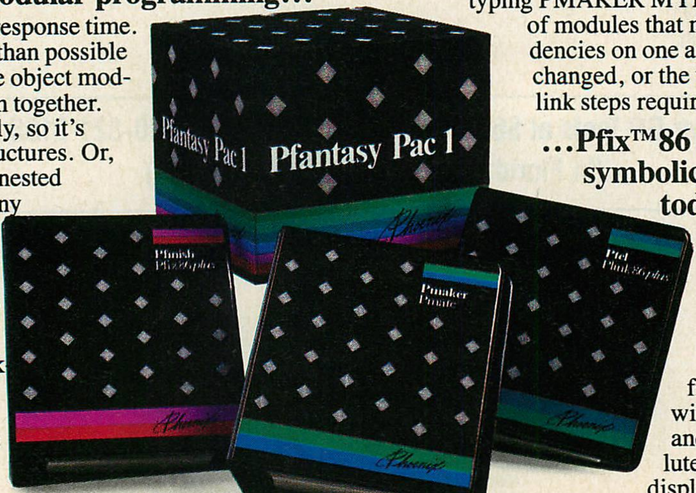


Buy Yourself A Pfantas
Just $~~12~~

Give your favorite programmer a holiday treat – Phoenix Pfantasy Pac1. Six of the best-engineered, highest performance programming packages on the market today. $1800 sold separately. $1295 as Pfantasy Pac1 with new software, and a unique on-line software update service. All at the one-time holiday price of $995. The perfect gift for the knowledgeable programmer. You'll get:

...Plink™86 Plus, our new linkage editor, adds a new dimension to modular programming...

Cache overlays to reduce response time. Link programs much larger than possible with standard linkers. Merge object modules so you can package them together. Reload overlays automatically, so it's easier to design program structures. Or, create up to 4,095 overlays, nested up to 32 levels, in one or many files, on one or many disks.



...Plus, Pmate,™ the text editor that works the way you do...

You can run Pmate in background while you perform another task, or call it up instantly while you're inside another program. With Pmate, you can reassign any command or key, so you can use keystrokes learned on other editors or make your keyboard Dvorak. And, you can assign complicated editing tasks to a keystroke, so you never have to re-key them.

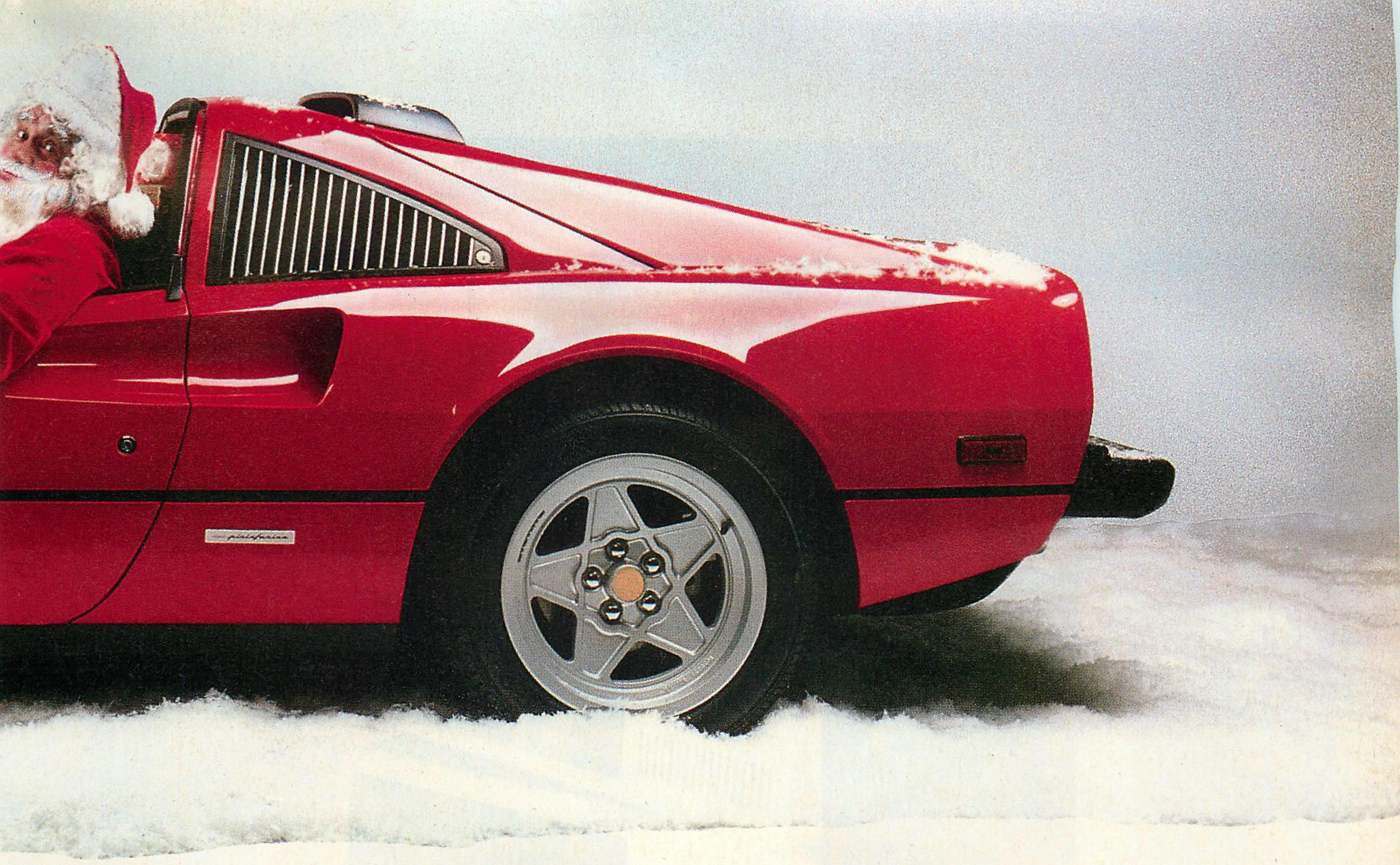
...Pmaker,™ the perfect program development manager...

Like UNIX™ Make and MMF, Pmaker updates and tracks modifications in your programs. With Pmaker, generating a final copy of your program is never more complicated than typing PMAKER MYFILE, regardless of the number of modules that make up MYFILE, their dependencies on one another, the number of modules changed, or the variety of compile/assemble/link steps required to generate the program.

...Pfix™86 Plus, the most advanced symbolic debugger on the market today...

With its adjustable window display of source file, disassembled object, data area, stack, breakpoint settings, CPU and coprocessor registers, Pfix takes the guesswork from debugging. You can debug without a listing, since you can see and enter symbolic names or absolute addresses in breakpoints, data displays, expressions, or with the in-line assembler. The entire 80xx and 80xxx series of

*Price effective through December 31, 1985.



y Pac™ For The Holidays.

5. $995.*

16-bit processors is supported, along with the user-assignable string and numeric variables, up to 100-step trace-back, and debug log to disk or printer. Breakpoints can be set in the source file window, whose display can be synchronized with the disassembled object during tracing; symbols can be added incrementally during the debug session; and the disassembly written to disk.

...Pfinish,™ the program that maximizes your program's efficiency...

Pfinish helps you turn your beta-test product into a software work of art. It analyzes your program or the entire operating environment during execution, and produces reports which tell you how much time was spent in each routine or interrupt, who called it, how many times, and much more. Wasteful, inefficient, or non-optimal areas of code become immediately apparent, whether they're in your code, the compiler library, or in the operating system routines themselves.

...Ptel™ gets the lead out of binary file transfer...

You get error-free file transfer and access to mainframes, minis, and micros. ASCII. XModem. Modem7. Telink and Kermit. MS-DOS® 2.x or 3.x. With Ptel. You can transfer 8-bit binary files over a 7-bit data path with Kermit. Masked file-name transfers with XModem. Or, transfer whole subdirectories with a single command using Telink. Ptel keeps track of the original file size and creation date. Ptel also offers a script language, backward scrolling, and the ability to handle DOS commands from inside the program.

CIRCLE NO. 205 ON READER SERVICE CARD

...And, Phoenix's new on-line software update service. Free to all registered Pfantasy Pac1 owners.

Pfantasy Pac1 software works in MS-DOS/PC DOS environments, is completely language-independent, and is available for the IBM® PC, XT,™ AT,™ and compatibles, the Wang® PC, the TI Business-Pro,™ and the Tandy® 2000.

So, spoil yourself a little. Get Pfantasy Pac1, and make your favorite programmer's dream come true.

For more information contact: Phoenix Computer Products Corporation, 320 Norwood Park South, Norwood, MA 02062. Or call: (800) 344-7200. In Massachusetts (617) 762-5030.

Programmers' Pfantasies™ by

Phoenix

Programmers' Pfantasies, Pfantasy Pac, Pmaker, and Pfinish are trademarks of Phoenix Computer Products Corporation. Pmate, Pfix86 Plus, Plink86 Plus, and Ptel are trademarks of Phoenix Software Associates Ltd. MS-DOS is a registered trademark of Microsoft Corporation. Wang is a registered trademark of Wang Laboratories, Inc. Tandy is a registered trademark of Tandy Corporation. UNIX is a trademark of AT&T Bell Laboratories. IBM is a registered trademark, XT and AT are trademarks of International Business Machines Inc. TI Business-Pro is a trademark of Texas Instruments Incorporated.

SEE US AT FALL COMDEX


parse(Item,Before,After,[Item,Result1,Result2]) :-
phrase(Item,Element1,Element2),
parse(Element1,Before,After1,Result1),
parse(Element2,After1,After,Result2).
parse(Item,[Element1|Remainder],Remainder,
[Item,Element1]) :-
word(Item,Element1).
phrase(s,np,vp).
phrase(np,d,n).
phrase(vp,v,np).
IBM

Programming in Logic

Part 1

With Prolog the user supplies only the facts; the computer finds the solution.

MICHAEL COVINGTON

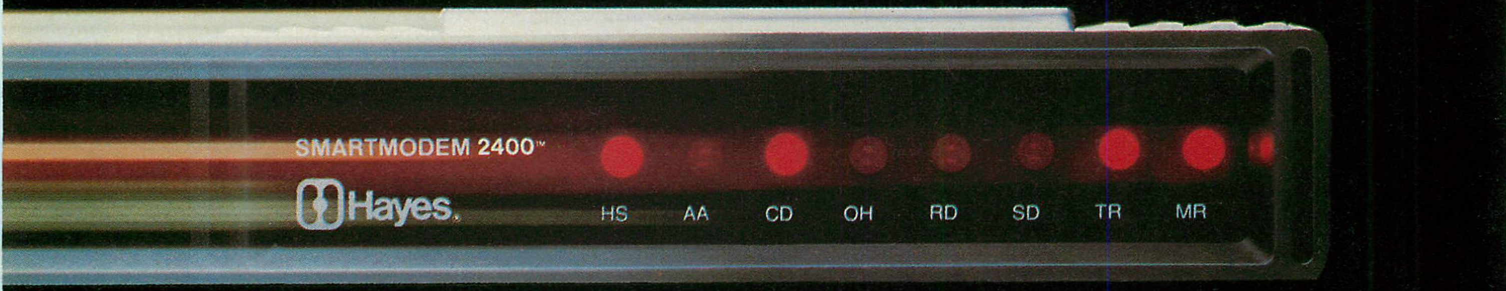
For decades, programming a computer has meant telling a computer how to solve a problem—that is, breaking the problem down into a series of steps that the computer is capable of following to some conclusion. However, the new programming language called Prolog may change all this. In Prolog, the programmer does not give the computer a set of steps to work through; rather, he merely provides a set of facts, and the computer determines on its own how to manipulate these facts in order to solve the particular problem.

Prolog programs, because they consist of facts rather than procedures, sometimes are referred to as databases. Prolog, however, is much more powerful than any conventional database program. For example, with Prolog the user need not define record formats or data fields; he simply adds information to the program as necessary. In fact, no distinction is made between program and data; inferred or derived facts are treated exactly as if they had been included explicitly in the database.

Prolog's power lies in its ability to infer facts from other facts. Traditionally, computers have been used either to calculate numerical information or simply to regurgitate information of other kinds. A computer provided with a set of numbers is able to calculate the average, which represents new information not entered into the computer by the user. If, however, the computer simply is given a set of names and addresses, it is unable to do anything but give them right back. It may give them back selectively or in a different order, but it cannot derive any new information from them. In Prolog, on the other hand, the user can give the computer non-numeric information and have it deduce additional non-numeric information on its own.

ILLUSTRATION • MARK TAUSS

Hayes sets the standard for personal computer communications. Again. Smartmodem 2400.™



The new fast mover from Hayes. The telecomputing leader. When it comes to communications products for personal computers, we're the leader! Hayes Smartmodem 1200™ set the industry standard for quality, reliability and performance.

Now our new, faster Smartmodem 2400 goes even further to lower telephone line costs and improve user productivity. So, at twice the speed of a 1200 bps modem, it quickly pays for itself in any high-volume communications operation.

Smartmodem 2400 provides a quick link to minis and mainframes. Both synchronous and asynchronous transmissions are supported by an advanced version of the well-known Hayes "AT" command set. You can download from the IBM mainframe at the home office. Send data to the mini upstairs. And guarantee accurate transmission with information services.

With worldwide communications in mind, Smartmodem 2400 was designed to meet CCITT international standards. It provides a fast, cost-effective way to transmit data between approved countries, even when they're continents apart.

New version of Hayes Smartcom II® communications software creates a complete telecomputing system with Smartmodem 2400. Our new Smartcom II, Version 2.1, is available for the IBM® PC and many popular compatibles. Smartcom II makes the most of Smartmodem's exceptional features, at the same time it makes communicating easy for you. And, if you're currently using an earlier version of Smartcom II, Hayes offers a $25 upgrade to Version 2.1.

So if you're involved with linking micros and streamlining corporate communications, rely on Hayes Smartcom II and Smartmodem 2400. The new communications standard. Again.

For information, see your authorized Hayes dealer or call us direct.

Hayes Microcomputer Products, Inc., P.O. Box 105203, Atlanta, GA 30348. 404/441-1617.

Smartmodem 2400
• Direct connect • Asynchronous and synchronous communications • Accommodates Hayes-compatible modems of slower speeds • Meets CCITT worldwide standards • Keyboard control of all communications parameters • High speed indicator • Voice/data capabilities • Call progress monitoring • Two-year limited warranty with optional four-year extended warranty available.

Smartcom II
• Hayes Verification and XMODEM protocols • Emulates DEC* VT52 and VT100/102 • Totally unattended operation • Voice/data capabilities.

Hayes®
Innovative products for enterprising people

CIRCLE NO. 155 ON READER SERVICE CARD

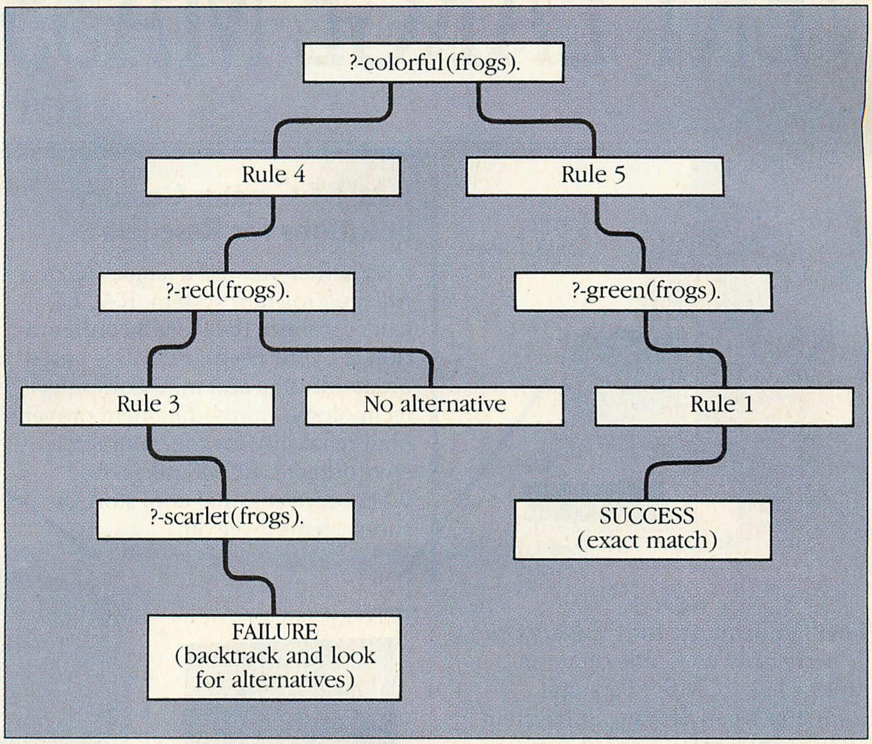
Hayes products now available under GSA Contract #GS00K85AGS5080.

FIGURE 1: *Comparison*

```
In English:

[1]  Frogs are green.
[2]  Roses are scarlet.
[3]  X is red if X is scarlet.
[4]  X is colorful if X is red.
[5]  X is colorful if X is green.

In Prolog:

[1]  green(frogs).
[2]  scarlet(roses).
[3]  red(X) :- scarlet(X).
[4]  colorful(X) :- red(X).
[5]  colorful(X) :- green(X).
```

Each line in English is translated to the corresponding line in Prolog. The symbol ":-" is read "if." Prolog lines 1-5, if entered into Prolog, would comprise a legal Prolog database.

FIGURE 2: *Search Tree for a Prolog Query*



Strictly speaking, the physical order of rules in the database is irrelevant to the search strategy. In practice, reordering of rules can change search speed markedly.

The name *Prolog* is taken from *pro*gramming in *log*ic. The language was developed in the early 1970s by Alain Colmerauer at the University of Marseilles. Prolog implements a subset of classical logic; it solves problems by means of a theorem-proving algorithm that attempts to derive a conclusion from a set of premises.

This theorem-proving algorithm operates by pattern-matching and backtracking. A query, or goal, is matched to inference schemas, or rules, in order to derive new goals; this enables Prolog to go through many steps of reasoning when working to solve a single problem. If one rule does not lead to a solution, Prolog backs up and tries another.

A SIMPLE QUERY

Before a Prolog program can be considered, some rules of syntactics must be noted. First, while *frogs* and *roses* are plural in English, each represents a single item as far as Prolog is concerned. Second, the **:-** symbol is read as *if* by Prolog; this, however, is not the same *if* as that of classical logic. Furthermore, any term that begins with a capital letter is recognized in Prolog as a variable; predicates and other language elements begin with a lowercase letter. In the example that is discussed below, *X* is a variable, but all other items, such as frogs, scarlet, and colorful, are not. Finally, when programming in Prolog, comments are enclosed in the symbols **/* */**, as they are in PL/1.

Figure 1 shows a Prolog program that consists of five rules, each of which defines a predicate, such as colorful, green, or scarlet, by naming the individuals to which it applies, such as frogs or roses. If this program is entered into the computer and followed by the question, "Are frogs colorful?", the computer's answer should be "yes," which is the logical inference to be made from rules 1 and 5. The question is asked by typing in the following query:

?-colorful(frogs).

Prolog begins by looking for a rule that will yield **colorful(frogs)** as a conclusion. Obviously, the first three rules will not suffice, because they assert that particular objects are green, scarlet, and red, respectively. If the variable *X* is set to equal frogs, then the first part of rule 4 matches the present goal exactly. The process of assigning values to variables is called *instantiation* or *unification*; thus, the query has now been unified with the left-hand side of rule 4.

As a result, the new goal becomes **red(frogs)**. Only rule 3, which begins with **red(*X*)** could possibly satisfy this new goal. After *X* is set to equal frogs (this *X* is not the same variable as the *X* in rule 4; they have the same name but different scopes), rule 3 can be applied; the new goal is **scarlet(frogs)**.

Prolog now has a problem. It can find no rule to pattern-match with **scarlet(frogs)**. Consequently, it backs up to **red(frogs)** and looks for a second way to satisfy that goal. Again it is stumped because only one rule begins with red, and that already has been tried. So Prolog backs up one more step to the original goal, **colorful(frogs)**.

Now Prolog finds an alternative. The first time through, **colorful(frogs)** was matched to rule 4; this time, it can be matched instead to rule 5. The new goal becomes **green(frogs)**, which matches rule 1 exactly; no new goal is generated, and the quest is over. Because all goals have been satisfied, the answer to the original question is "yes," or, in Prolog parlance, the original query has succeeded. Figure 2 shows the whole process in the form of a search tree. The complete dialogue between the user and the machine looks like the following:

?-colorful(frogs).
yes

If the query had failed, Prolog would have answered "no."

Notice that the order in which the rules were tried did not affect the result. Reordering the rules might increase or decrease the amount of time necessary for the search, but it will not affect the outcome. This stands in stark contrast to conventional programming languages that necessarily require statements to be executed in a particular order. It also leads to a modular style of programming; theoretically, every rule

FIGURE 3: *Sample Run of Listing 1*

```
A>prolog86

+------------------------------------------+
| MS-DOS Prolog-1             Version 2.2   |
| Copyright 1983      Serial Number: 000749 |
| Expert Systems Ltd.                       |
| Oxford, U.K.                              |
+------------------------------------------+


?- consult(CAR).
Type 'start.' to begin.
CAR consulted.
?- start.

Tiny sample expert system / M. Covington 1985

This program diagnoses why a car won't start.
Answer all questions with 'yes.' or 'no.'
(lower case, with period, without quotes).

When you first started trying to start the car,
did the starter crank the engine normally?
yes.

Does the starter crank the engine normally now?
no.

The battery has apparently become drained during
your attempts to start the car.  Recharging or
jump-starting will be necessary.  However, there
is probably nothing wrong with the battery
itself.

Is fuel being delivered to the carburetor?
no.

Check whether there is fuel in the tank.
If so, check for a clogged fuel line or filter
or a defective fuel pump.

A>
```

This is a screen dump of an actual Prolog session of the program EXPERT.PRO. The "consult" command loads a database file called CAR containing the rules for the expert system.

FIGURE 4: *Recursive Factorial Program*

```
In Pascal:     function factorial(x:integer):integer;
               begin
                 if x=0
                   then factorial:=1
                   else factorial:=x*factorial(x-1)
               end;

In LISP:       (DEFUN FACTORIAL (X)
                  (COND
                    ((= X 0)  1)
                    (T  (* X (FACTORIAL (- X 1))))))

In Prolog:     factorial(0,1).

               factorial(N,Fact) :-
                  M is N-1,
                  factorial(M,FactM),
                  Fact is N*FactM.
```

The second of the two Prolog rules continues to unify with itself until N is decremented to 0, at which time the first rule (which is nonrecursive) is able to satisfy the goal.

FIGURE 5: *Recursive Function to Concatenate*

```
In LISP:    (DEFUN APPEND (X Y)
               (COND
                 ((NIL X)  Y)
                 (T   (CONS (CAR X) (APPEND (CDR X) Y)))))

In Prolog:  append([],X,X).

            append([X|Y], Z, [X|NewTail]) :-
               append(Y, Z, NewTail).
```

The first two Prolog parameters are the lists to append; the third is the list that append returns. The LISP routine is a function while the Prolog routine is a procedure.

in a Prolog program can be understood in isolation from all the others.

FILLING IN THE BLANKS

If given a query with a variable in it, Prolog will attempt to make the query succeed by assigning the variable an appropriate value. In other words, it will fill in the blanks. The question

?-colorful(*X*).

is interpreted by Prolog as meaning, "Is there anything that is colorful?" The answer is "yes" because frogs and roses are colorful. If the rules are stored in the order shown in figure 1, the solution **X = roses** is the first to be located, and Prolog replies thus:

X = roses
yes

Because Prolog syntax has no standard, some implementations may not print out the value of *X* automatically as in the above example. In these instances a compound query such as

?-colorful(*X*),write(*X*),nl.

must be issued. This command query tells Prolog to find a value of *X* such that **colorful(*X*)** succeeds, then to write out the value and start a new line. The variable *X* retains its value in each of the three subgoals that make up the compound goal.

This still produces only one answer. To obtain all possible answers, the built-in pseudo-predicate **fail** should be used. **Fail** instructs Prolog that, even if it has succeeded up to a certain point, it should backtrack and try to find another solution. Thus, the query

?-colorful(*X*),write(*X*),nl,fail.

produces this reply from Prolog:

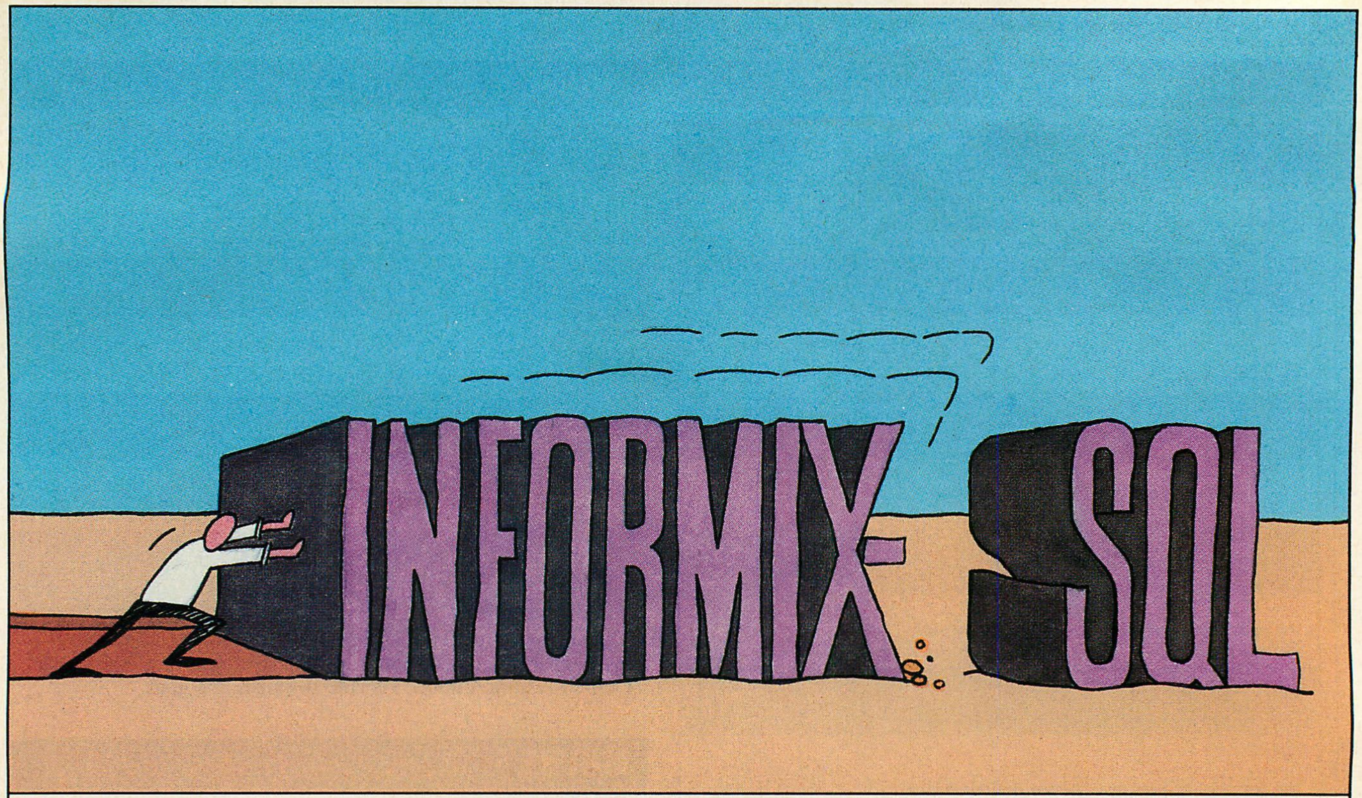
roses
frogs
no

The list of answers ends with "no" because the overall compound query eventually fails when Prolog runs out of ways to backtrack, even though solutions have been found along the way.

EXPERT SYSTEMS

Because it automatically draws inferences, Prolog often is used to write expert systems (programs that give advice to humans) and has been chosen as the basis of many important artificial intelligence projects, including the Fifth Generation Project in Japan.

Listing 1, EXPERT.PRO, is an example of a small expert system that provides advice on why a car will not start; figure 3 is a sample run of this program. The CONSULT (CAR) expression instructs Prolog to load the file CAR and add its facts and rules to the database in memory. Not very much automotive expertise can be encoded in a program that is this short, but the example at



How we improved Structured Query Language.

Actually, we didn't change a thing.

We just combined it with the best relational database management system.

Introducing INFORMIX®-SQL.

It runs on either MS™-DOS or UNIX™ operating systems. And now with IBM's SQL as part of the program, you can ask more of your database. Using the emerging industry-standard query language.

To make your job easier, INFORMIX-SQL comes with the most complete set of application building tools. Including a full report writer and screen generator. Plus a family of companion products that all work together.

Like our embedded SQLs for C and COBOL. So you can easily link your programs with ours. File-it!,™ our easy-to-use file manager. And C-ISAM,™ the de facto standard ISAM for the UNIX operating system. It's built into all our products, but you can buy it separately.

And when you choose RDS, you'll be in the company of some other good companies. Computer manufacturers including AT&T, Northern Telecom, Altos and over 60 others. And major corporations like Anheuser Busch and The First National Bank of Chicago.

Which makes sense. After all, only RDS offers a family of products that work so well together. As well as with so many industry standards.

So call us for a demo, a manual and a copy of our Independent Software Vendor Catalog. Software vendors be sure to ask about our new "Hooks" software integration program. Our number: 415/322-4100.

Or write RDS, 4100 Bohannon Drive, Menlo Park, CA 94025.

And we'll show you how we took a good idea and made it better.

RDS™

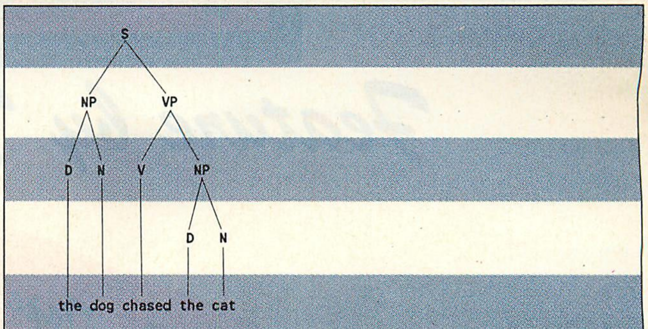
RELATIONAL DATABASE SYSTEMS, INC.

FIGURE 6: *Grammar to Generate Sentences*

```
S  -->  NP  VP   (sentence = noun phrase + verb phrase)

NP -->  D   N    (noun phrase = determiner + noun)

VP -->  V   NP   (verb phrase = verb + noun phrase)

D  -->  the      (determiners)
D  -->  a

N  -->  boy      (nouns)
N  -->  dog
N  -->  cat

V  -->  saw      (verbs)
V  -->  watched
V  -->  chased
```

A useful formal grammar for a modern language would require hundreds of thousands of entries in order to enable the successful parsing of arbitrary text.

FIGURE 7: *Structural Description of a Sentence*




```
[s,[np,[d,the],[n,dog]],[vp,[v,chased],[np,[d,the],[n,cat]]]]
```

A Prolog parser must be able to reconstruct the structural description of a sentence after it has been given the formal grammar. The actual output of the parser is a list.

least gives an idea about what expert system programming can entail.

Almost all of the rules in this program introduce compound goals. That is, if a query unifies with a rule, it will succeed only in the case where all of the parts of the compound goal succeed in the order given. This is as near as Prolog comes to step-by-step procedural programming. In many cases, the compound goal prints several lines of text with the help of **writeln**, a predicate defined at the bottom of the listing that prints a message without its quotes then starts a new line.

In several places the **cut** operator, written !, is used to suppress backtracking. **Cut** instructs Prolog not to backtrack past a certain point, or, if backtracking is unavoidable, to jump back all the way to the goal that originally invoked the rule containing the cut.

The program is invoked by querying the goal **start**. After printing a few lines of introductory text, start invokes the goal **try-all-possibilities**, which has two definitions. The first definition instructs Prolog to identify a defect and print the appropriate explanation, then to backtrack and try again. When Prolog has run out of ways to backtrack, the second definition of **try-all-possibilities** tells it to **halt**, to exit Prolog, and to return the user to DOS.

The definitions of **defect_may_be** state the observations that eventually will lead the computer to a particular diagnosis. The observations, described by predicates such as **starter_is_ok**, are obtained as a result of questions directed to the user. Whenever an observation is obtained, it is added to the database; if the same information is needed later, the question will not be repeated. The pseudo-predicate **retract** deletes **starter_is_ok(unknown)** from the database; **asserta** adds **starter_is_ok (yes)** or **starter_is_ok(no)** to the database, depending on the user's response.

The user need not construct a flowchart or logic tree in order to write a program such as the one that diagnoses why a car will not start. Instead, he must simply list some possible causes along with the observations pointing to each. The overall logical structure of the program is handled by Prolog. This feature becomes crucial when working with large expert systems in which the overall picture is too big for any human being to comprehend. With Prolog, the programmer is able to judge the correctness of any small part of a system even if he does not understand the entire program.

The user need not construct a flowchart or logic tree in order to write a program to diagnose, for example, why a car will not start. The logical structure is handled by Prolog.

LIST PROCESSING

Like LISP, Prolog makes extensive use of recursion. It accomplishes this by instructing a rule to invoke a new goal that will unify with that same rule. Figure 4 shows the familiar recursive factorial algorithm that is expressed in Pascal and LISP as well as in Prolog. Translated into English, the Prolog example means the following: "The factorial of 0 is 1. In all other cases, the factorial of *N* is Fact if *M* equals *N*-1, the factorial of *M* is Fact*M*, and the product of *N* and Fact*M* is Fact."

In Prolog, lists are written as sequences of symbols between square brackets. A list can be taken apart either by pattern-matching it to a list of variables (for example, *[a,b,c]* will match *[X,Y,Z]*) or by matching it to a structure of the form *[X:Y]*, where *X* is the first element of the list, and *Y* is a list consisting of all the other elements. An empty list is written [ ]; a one-element list will match *[X:Y]* but will set *Y* equal to the empty list.

LISP algorithms are easy to translate into Prolog as long as they do not include the PROG, GO, or LOOP constructs, none of which has an exact Prolog equivalent. Figure 5 shows the standard LISP APPEND function (which concatenates two lists) and its translation into Prolog. The COND operator of LISP is rendered into Prolog by splitting the function into two rules; the second one is invoked only if the first one does not match the goal. Note that Prolog goals are not function calls; the only way they can return values other than success or failure is by instantiating variables passed to them as arguments.

The combination of backtracking, recursion, and list processing makes Prolog ideal for writing parsers, programs that can recognize the syntax of a language, be it a programming language or a human language.

The first step in constructing any parser is to give an exact description of the language to be parsed, usually by

means of a formal grammar. Figure 6 shows a formal grammar for a tiny subset of English. Every sentence generated by this grammar has a corresponding structural description that shows which rules were used to generate it; figure 7 gives an example of a structural description. A parser reads a sentence and then reconstructs the structural description, thereby making clear the role that each word plays in the structure of the sentence as a whole.

A parser should have a hierarchical structure that is similar to that of the grammar of the language being parsed. For example, this parser, because it is parsing English, should look first for a noun phrase, which is the combination of a determiner and a noun. To find a determiner, the parser looks systematically for each of the determiners in its vocabulary (*the* and *a*); to find a noun, it looks for any of the nouns in its vocabulary. The backtracking mechanism of Prolog is ideally suited for a task organized in this manner.

Next, the parser should look for a verb phrase. If it finds both the noun phrase and the verb phrase and no words are left over at the end, the parser concludes it has found a sentence.

Listing 2 shows a parser, written in Prolog, for the sample grammar used here. It parses sentences written in Prolog list format; code could be added, however, to make it accept sentences written in a more normal format.

Most of the work is done by the predicate parse, which takes four arguments: the constituent being looked for; the list of words to be parsed; the same list after the desired constituent has been picked off; and the structural description of the constitutent being parsed. When parse is invoked, its first two arguments should already contain values; the latter two are variables that parse will instantiate itself.

If the constitutent being sought is a two-element phrase, parse calls itself twice, each time seeking one of the two elements. If, on the other hand, the constituent being looked for is a word, the second definition of parse takes over, and that word is simply picked off the input list and added to the structural description in the proper place.

To parse the sentence, *The cat saw the dog.*, Prolog must be queried with:

```
?-parse(s,[the,cat,saw,the,dog],[ ],X).
```

This instructs Prolog to parse a sentence that starts with *the cat saw the dog* and ends with nothing left over, then to assign that sentence's structural description to the variable *X*. Prolog answers this query by returning this structural description to the user:

```
X=[s,[np,[d,the],[n,cat]],[vp,[v,saw],[np,
   [d,the],[n,dog]]]]
yes
```

This parsing program is surprisingly short. A similar parser in Pascal would take five or six pages of code. Even in LISP the program might be twice the size of the Prolog version.

Moreover, this parser can perform a task that parsers written in other programming languages are unable to do. Like the Prolog language in which it is written, it is able to fill in the blanks. If the user asks it to parse a sentence that has variables in place of one or more words, the parser will supply appropriate words for each of these variables automatically. If queried

```
?-parse(s,[the,cat,X,the,boy],[ ],Y).
```

this parser will reply:

```
X=saw
Y=[s,[np,[d,the],[n,cat]],[vp,[v,saw],[np,
   [d,the],[n,boy]]]]
yes
```

This ability makes Prolog parsers useful in teaching foreign languages to humans. The parser can produce sample sentences randomly or according to a preset pattern, just as random number generators generate equations for addition or multiplication drills.

TWO SYNTAXES FOR PROLOG

Most Prolog interpreters use the syntax described by W. F. Clocksin and C. S. Mellish in their book *Programming in Prolog*. This is sometimes called Edinburgh syntax because that is where Clocksin and Mellish implemented their Prolog interpreter. The accompanying article uses their syntax. The older tradition, however, writes Prolog clauses as LISP lists. Therefore, instead of

```
green(frogs).
colorful (X) :- green (X).
```

the programmer would write:

```
((GREEN FROGS))
((COLORFUL X) (GREEN X))
```

Note that one set of parentheses is placed around the whole rule and a second set is placed around each clause within the rule.

This LISP-like syntax originated in experimental Prolog interpreters that were written in LISP. Of the Prolog interpreters available for the PC, only Micro-Prolog uses this syntax.

The LISP-like syntax is much easier for the computer to parse. However, it has one big disadvantage: it is unable to evaluate arithmetic expressions. Instead of *X* is *Y+Z*, the programmer has to write (SUM *Y Z X*). Consequently, the Micro-Prolog translation of a complex algebraic formula can be quite unreadable.

Another difference concerns variable names. In Clocksin and Mellish's Edinburgh syntax, any word beginning with a capital letter is recognized as a variable. In Micro-Prolog, on the other hand, variables can have the form *X*, *Y*, *Z*, *x*, *y*, *z*, *X*1, *Y*1, *Z*1, *x*1, *y*1, *z*1, and so on, and the interpreter can even change variable names in order to put them in the canonical sequence.

At first sight, Edinburgh syntax seems the better of the two. However, LISP-like syntax is simpler and can be easier to learn. In addition, it saves the interpreter a great deal of work. As a result, Micro-Prolog is one of the fastest Prolog interpreters available. Until very recently, no Clocksin and Mellish Prolog interpreters could boast comparable speeds.

—*MC*

LIMITATIONS OF PROLOG

Prolog has some disadvantages, one of which is the frequent need for an awkward mix of procedural and nonprocedural styles. Although Prolog's nonprocedural approach is ideal for automated reasoning, it is almost incapable of handling algorithms that must be performed in a step-by-step manner. Even such simple procedures as prompting a user for a series of data items require awkward Prolog constructions involving long compound queries and the frequent use of **cut**. Many Prolog programmers believe that the ideal computer language would combine the reasoning power of Prolog with the ability to call subroutines defined in a more procedural language such as LISP.

Prolog is not able to express any negative conclusions or embedded conditionals. For example, it cannot say "*X* is not a frog if *X* has fur," or even "Kermit is not a frog." Nor is Prolog able to express a definition that contains more than one *if*, such as "*X* is fragile if *X* breaks if *X* is dropped."

PERSONAL COMPUTER PHOTOCOPIER

Have you ever wished that there was a machine for your PC that could scan your photographs, artwork or documents just like a photocopier? And was as easy to use and understand? As well as copy and store in color or black-and-white onto your hard drive for editing?

Well, there is such a machine now. It's called the **SpectraFAX** digital photocopier. We got tired of trying to use digitizing tablets, cameras, mice and everything else that has been invented. The **SpectraFAX** machines will scan any and all of your art, photos, forms or text for storage in your PC. Images that you can cut-and-paste into your word processing or database programs. Our **SpectraFAX 200** will scan and store your color images for editing and printing on your color printer at resolutions up to 200 dots per inch. The **SpectraFAX DS-200** will do the same for you in black-and-white. Then the **SpectraFAX Graphics Editor** software that we bundle with the scanner will allow you to cut-and-paste your images, rotate them, enlarge and reduce whatever pictures you choose.

That's not all. Our digital photocopiers are based on open-architecture design, which means that as new uses for this technology develop, add-on cards can enhance our **SpectraFAX** digital photocopier. But we didn't want to wait for third-party vendors, so we invented two optional boards of our own: the **SpectraFAXimile Card** and the **SpectraFAX TEXreader**. The **SpectraFAXimile card** takes advantage of the 200 dots per inch resolution of the scanning device to turn it into a facsimile machine. The **TEXreader** board is a revolution in itself — full-scale OCR (optical character recognition) to read typewritten pages with the **SpectraFAX** machine into your word processor without having to re-type what somebody else has already typed.

What's most amazing about the **SpectraFAX** product line, though, are the prices. Our basic black-and-white digital photocopiers cost about the same as a regular office copier (which you may never use again!) and our color scanners cost very little more. Because we know high tech has to be affordable.

• IBM is a registered trademark of International Business Machines, Corp.
• SpectraFAX, SpectraFAX 200, DS-200, SpectraFAX Graphics Editor, SpectraFAXimile & TEX reader are trademarks of SpectraFAX Corp.

SpectraFAX Corp.

2000 Palm Street S. • Naples, Florida 33962 • (813) 775-2737

CIRCLE NO. 194 ON READER SERVICE CARD

In some important cases, Prolog goes into an infinite loop instead of generating a solution that follows logically from the available facts. For example, the following two rules

```
/* 1 */ sibling(harpo,groucho).
/* 2 */ sibling(X,Y) :- sibling(Y,X).
```

instruct Prolog that "Harpo is the sibling of Groucho. For any *X* and *Y*, *X* is the sibling of *Y* if *Y* is the sibling of *X*." If the computer is queried.

```
?-sibling(harpo,fred).
```

it should try to determine whether Harpo is the sibling of Fred. First it tries rule 1, which does not work; then it tries rule 2, which does work, and the new goal becomes **sibling(fred,harpo)**. Again, rule 1 does not match, but rule 2 does. When rule 2 is applied a second time, however, the query again becomes **sibling(harpo,fred)**, returning Prolog to where it started. Loops such as this result from any attempt to express symmetric (commutative) or transitive relations, as well as from any pair of rules that form a biconditional (*A* if *B*; *B* if *A*). Yet rules of all these types are needed if Prolog is to encode propositions that occur frequently in human thought. Prolog did not simply overlook the need to incorporate these features; good algorithms for handling such situations did not exist when Prolog was first developed. Only now are they finally coming into being.

THE NEXT GENERATION

In spite of its limitations, Prolog has been adopted by the Fifth Generation Project and similar supercomputer projects because of its nonprocedural nature. Traditional computers execute instructions serially, and languages such as FORTRAN and Pascal are ideally suited to them. The computers of the future, however, will rely heavily on parallel processing in order to accomplish more work in less time.

Computers in use today operate as fast as the laws of physics will allow. On the IBM PC, one clock cycle, the shortest possible step into which computations can be broken, takes about 210 nanoseconds to execute. The fastest mainframe computers have clock cycles of about 20 nanoseconds, which is only 10 times faster than the PC.

The mainframes greatly outperform the PC because their more sophisticated processors can accomplish more during the time of each clock cycle. However, shortening the clock cycle itself probably would not result in increased speed of the machine.

The problem is that, in 20 nanoseconds, an electrical signal moving at the speed of light travels only 20 feet. If the clock cycle were much shorter, signals simply could not move from one end of the machine to the other in time to perform their functions. Miniaturizing the machine might make a 2-nanosecond clock cycle a workable possibility. That, however, is probably the limit because, in addition to moving signals around, a computer must process them, and the response time of the components has to be taken into account.

The next generation of computers must rely on making independent processors work simultaneously. While programs written in Pascal or FORTRAN would be difficult to adapt for use with such machines, Prolog is ideally suited for such an application. Its structure allows two kinds of parallel execution, or-parallelism and and-parallelism.

Of these two, or-parallelism is the easiest to implement because it tests for alternative possible solutions at the

The next generation of computers must rely on making several independent processors work simultaneously; Prolog is ideally suited for such machines.

same time. That is, whenever the current goal can be unified with more than one rule, all of the resulting new goals are pursued simultaneously on separate processors. If one processor finds a solution, and the query does not call for an exhaustive search, then that processor signals all the others to stop the search. Alternative branches of the search tree do not pass information to each other; thus, no interaction is possible between the various processors.

And-parallelism means that all the goals in a compound query are tried at the same time rather than in sequence. For example, if the computer is asked, "Are frogs green?" and the program contains the rule, "*X* is green if *X* is slimy and *X* is aquatic," then the two new queries, "Are frogs slimy?" and "Are frogs aquatic?", will be tested simultaneously on different processors. The implementation of and-parallelism requires modification to the Prolog language because compound goals often are used to express actions that must be done in sequence, as well as to pass information from one goal to another in the form of values assigned to variables. At the very least, an and-parallel version of Prolog needs a way to suppress parallelization when a series of queries must be evalutated in sequence.

Prolog is not the last word in nonprocedural languages, but it is an important first step. Research is in progress to develop a much better nonprocedural language with Prolog as its foundation, and then to implement that language on a fast, parallel computer. When this is accomplished, computers will be used much more frequently for nontraditional purposes. In 20 years, expert systems may be as common as encyclopedias, computers may understand English, and the present procedural programming languages may seem as out-of-date as mechanical adding machines seem today.

REFERENCES

Barr, Avron, and Feigenbaum, Edward A., editors. *The Handbook of Artifical Intelligence*. 3 vols. Stanford: HeurisTech; Los Altos: William Kaufmann, 1981-82.

Chomsky, Noam. *Syntactic Structures*. The Hague: Mouton, 1957. (Seminal work on formal grammar of English.)

Clark, K. L., and McCabe, F. G. *Micro-Prolog: Programming in Logic*. Englewood Cliffs, New Jersey: Prentice-Hall, 1984. (Concentrates on SIMPLE, a user interface for Micro-Prolog that makes Prolog look more like English but deprives it of its simplicity.)

Clocksin, W. F., and Mellish, C. S. *Programming in Prolog*. Second edition. Berlin: Springer-Verlag, 1984.

Covington, Michael A. "Eliminating unwanted loops in Prolog." *ACM SIGPLAN Notices* 20.1 (January 1985), pp. 20-26.

Kowalski, Robert. *Logic for Problem Solving*. Amsterdam: North-Holland, 1979.

Moto-Oka, Tohru, editor. *Fifth Generation Computer Systems*. Amsterdam: North-Holland, 1982.

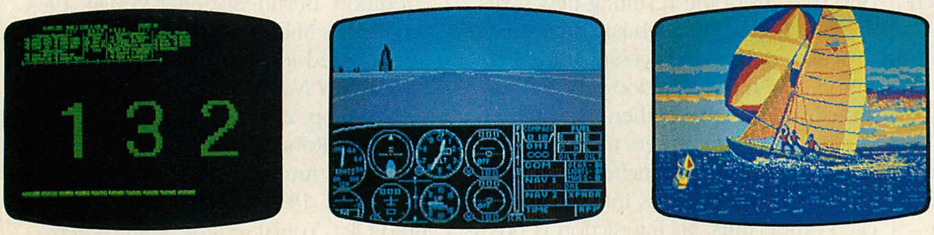
This is the first article of a two-part series. Part 2, appearing in the January 1986 issue, will review all major Prolog interpreters for the IBM PC.

Michael Covington is a research associate of the Advanced Computational Methods Center at the University of Georgia. He is working to implement a Prolog-based language on a CYBERPLUS parallel supercomputer.

DOES YOUR HI-RES GRAPHICS BOARD SHOW YOU ONLY PART OF A SPREADSHE

When it comes to printing out spreadsheets, most IBM®-compatible graphics boards come up short.

But not the Genoa Spectrum™ Color/Mono/Graphics Board.



The Spectrum Color/Graphics Board is the most compatible board you can buy. Which means it runs the most software. Including Flight Simulator,™ Lotus® 1-2-3,™ Symphony,™ SideKick,™ PC Paint,™ and much, much more.

It goes all the way to 132 columns. The largest text display of any graphics board you can buy. There's even a special driver that'll show you 132 columns of Lotus, which normally only has 80.

And together with your 132-column printer, the Spectrum can also print every one of those 132 columns. So you can see more of what you're doing. On the screen. And on paper.

But the best part is, you don't have to pay more to do it. Here's why:

Up until now, if you wanted both graphics and color, you had to buy two boards. One for hi-res mono graphics, one for color.

But the Genoa Spectrum puts color and graphics *on the same board.* So you save yourself a slot. And about $250.

You can also save yourself the price of a color monitor. Because Spectrum will emulate your color software in monochrome. And for your monochrome software there's a 720 x 348 hi-res mode.

So either way, you're covered. Spectrum is compatible with all color and monochrome monitors. So you'll always get 16 beautiful colors in color. Or 16 shades of monochrome in mono.

For all the technical specs or the Genoa Spectrum dealer nearest you, just call us at 408-945-9720. Or write Genoa Systems Corporation, 73 E. Trimble Road, San Jose, CA 95131.

The Genoa Spectrum. It's the only board you'll ever need for color, graphics, and full size output.

Anything else is either too much money or incomple

Genoa

We make IBM PCs better.

CIRCLE NO. 209 ON READER SERVICE CARD

LISTING 1: EXPERT.PRO

```
start :-
   nl,
   writeln('Tiny sample expert system / M. Covington 1985'),
   nl,
   writeln('This program diagnoses why a car won''t start.'),
   writeln('Answer all questions with ''yes.'' or ''no.'''),
   writeln('(lower case, with period, without quotes).'),
   !,
   try_all_possibilities.

try_all_possibilities :-
   /* The 'fail' at the end causes this predicate to  */
   /* backtrack through all possibilities.            */
   defect_may_be(Diagnosis),
   explain(Diagnosis),
   fail.

try_all_possibilities :-
   /* If all paths through the previous definition  */
   /* have been exhausted, leave Prolog.            */
   halt.

explain(wrong_gear) :-
   nl,
   writeln('Check that the gearshift is set to Park or'),
   writeln('Neutral.  Try both Park and Neutral and see if'),
   writeln('either of them works; try jiggling the gear'),
   writeln('shift lever.').

explain(starting_system) :-
   nl,
   writeln('Check for a defective battery, voltage'),
   writeln('regulator, or alternator; if any of these is'),
   writeln('the problem, charging the battery or jump-'),
   writeln('starting may get the car going temporarily.'),
   writeln('Alternatively, the starter itself may be'),
   writeln('defective.').

explain(drained_battery) :-
   nl,
   writeln('The battery has apparently become drained during'),
   writeln('your attempts to start the car.  Recharging or'),
   writeln('jump-starting will be necessary.  However, there'),
   writeln('is probably nothing wrong with the battery'),
   writeln('itself.').

explain(fuel_system) :-
   nl,
   writeln('Check whether there is fuel in the tank.'),
   writeln('If so, check for a clogged fuel line or filter'),
   writeln('or a defective fuel pump.').

explain(ignition_system) :-
   nl,
   writeln('Check the spark plugs, cables, distributor,'),
   writeln('coil, and other parts of the ignition system.'),
   writeln('If any of these are visibly defective or long'),
   writeln('overdue for replacement, replace them; if this'),
   writeln('does not solve the problem, consult a mechanic.').

defect_may_be(drained_battery) :-
   starter_was_ok(yes),
   starter_is_ok(no).

defect_may_be(wrong_gear) :-
   starter_was_ok(no).

defect_may_be(starting_system) :-
   starter_was_ok(no).

defect_may_be(fuel_system) :-
   starter_was_ok(yes),
   fuel_is_ok(no).

defect_may_be(ignition_system) :-
   starter_was_ok(yes),
   fuel_is_ok(yes).

starter_was_ok(unknown).

starter_was_ok(X) :-
   retract(starter_was_ok(unknown)),
   nl,
   writeln('When you first started trying to start the car,'),
   writeln('did the starter crank the engine normally? '),
   read(Reply),
   asserta(starter_was_ok(Reply)),
   !,
   X = Reply.

starter_is_ok(unknown).

starter_is_ok(no) :-
   starter_was_ok(no).

starter_is_ok(X) :-
   retract(starter_is_ok(unknown)),
   nl,
   writeln('Does the starter crank the engine normally now?'),
   read(Reply),
   asserta(starter_is_ok(Reply)),
   !,
   X = Reply.
   fuel_is_ok(unknown).

   fuel_is_ok(X) :-
     retract(fuel_is_ok(unknown)),
     nl,
     writeln('Is fuel being delivered to the carburetor?'),
     read(Reply),
     asserta(fuel_is_ok(Reply)),
     !,
     X = Reply.

   writeln(X) :-
     write(X), nl.

   ?- writeln('Type ''start.'' to begin.').
```

LISTING 2: PARSER.PRO

```
parse(Item,Before,After,[Item,Result1,Result2]) :-
    phrase(Item,Element1,Element2),
    parse(Element1,Before,After1,Result1),
    parse(Element2,After1,After,Result2).

parse(Item,[Element1|Remainder],Remainder,[Item,Element1]) :-
    word(Item,Element1).

phrase(s,np,vp).
phrase(np,d,n).
phrase(vp,v,np).

word(d,the).
word(d,a).

word(n,boy).
word(n,dog).
word(n,cat).

word(v,saw).
word(v,watched).
word(v,chased).
```

NEW STUFF!

IF YOU PROGRAM IN 'C' DON'T TURN THE PAGE!

PADLOCK

NEW LOW COST VERSION

Protect Any Program Using Anyone's Diskettes

You are a program developer seeking a low-cost method to protect your program from piracy? You are with a prominent company fearful of embarrassing lawsuits should employees copy programs illegally?

We have a devilishly clever solution for the two of you we call PadLock™. It gives you the keys to lock up any program you want—yours or someone else's— without buying a stockpile of expensive fingerprinted disks.

PadLock is software. One of its programs formats any manufacturer's disk, embedding your secret code in sectors the operating system will ignore. Another program installs this code in any program files you name, whether YourProg or WordStar®m, or any .COM or .EXE files. From then on there's no picking the lock without the combination: the coded software together with a coded disk.

So it's not despised copy prevention. Protected programs can be copied anywhere, to hard disks, to backup diskettes. It's a lock and key scheme. A protected program will just lock up at load time without a keyed diskette (possibly its own) in one of the drives. The key disk may then be removed to free the drive.

PadLock is economical. You buy software that fingerprints disks for as little as $2.50 each, not costly fingerprinted disks you may never use at maybe $8.00 each.

Tell you what. PadLock keeps track of how many disks it has formatted. Try it out. Format up to 5 test disks. If PadLock doesn't suit you, return it within 30 days for full refund.

Incidentally, don't look elsewhere. PadLock is available only from PC Brand.

Product:	Our Price:
M0030 Formats 30 disks	$ 99
M0100 Formats 100 disks	$250

BRIEF

OUTSTANDING FOR ANY LANGUAGE

Is Anything But. A Whopper of an Editor

Dvorak in *Infoworld* said it for us: "the word is getting around...that Brief, The Programmer's Editor is simply the best text editor you can buy." With a name that belies its thoroughness, Brief™ has every feature you've ever contemplated for your editor-in-chief. Here goes:

Text, from keyboard or files, is housed in multiple buffers, and scrolled through one or more windows you open, close, resize. A text buffer may be called to different windows to view two areas at once. A change in one changes both. Text blocks may be marked for printing, writing to files, movement to scrap buffers for cut and paste into other buffers, or deletion, with as many "undo" levels as you want.

To find and fix, Brief has text search abilities rivaling "grep," with wild cards for matching, indifference to intervening characters, acceptance of character ranges, even multiple choice of patterns and their replacements.

If you use Lattice, C86™, or Wizard, and have 320k, you can compile your C program without ever leaving Brief. In fact, Brief will then find the lines with errors, and march you through the text for repairs.

On to macros! Parts of Brief were written with its own Lisp-like macro language. It has structure, 32-character variable names, conditional execution, loops, and you can actually read it! Nothing like the hieroglyphs we've seen elsewhere. A compiler comes with your copy, and prewritten macros in source to give you the flavor. Among the notables: one automatically indents C text; another turns Brief into a word processer with margin setting and wraparound.

There's a bulletin board with public domain macros contributed by a mushrooming cult of macromaniacs, and we've even hearing of megamoney prices for magnificance.

Then tnere's ... but, really, we're out of space. Who named this product, anyway?

Product Code: U0590(	Our Price:
List Price: $195	$CALL
Needs 192k	

FIRSTIME

Assistant Programmer for Hire. Real Cheap.

Not interested in interpreters? Then explore the other fast lane to coding: language-specific editors. Among them, FirsTime™ is the strongest C editor we have seen, with many more commands and built-in utilities.

FirsTime asks what program structure you want and sets it up for you. It could be "main", or "function", or "if-else" or "while" or any component of C. Ask for it and FirsTime fills in its skeleton, tending to semi-colons and curly braces.

It puts placeholders into these structures to prompt you to enter workable code—the initializers, conditionals and counters which govern an "if", for example. It checks that variables you enter have been declared, and highlights errors until fixed. Material can be pulled into ten buffers for movement elsewhere. Your last delete is held in one of them for "undoing". There's search and replace, writing of any marked block to a satellite file or the printer.

FirsTime can even transform one type of structure into another, such as "for" into "while" or "while" into "do".

Movement through the screen highlights the successive logical blocks of your program so that its structure is always apparent. And FirsTime can hide different levels of code detail to show you only the outer framework of your program.

FirsTime saves keystrokes to add speed, indeed, but more important, it lets you think at a higher level and makes a big contribution toward error-free code.

Product Code: E0460	Our Price:
List Price: $295	$229

LMK

A Unix-like "Make" Makes Light of System Building in Any Language

If you have ever built a complex system, you know the time loss and tedium of recompiling, rebuilding libraries and relinking modules because a snippet or two of code has changed. Batch files are no answer. You need batches of them to avoid redoing everything indiscriminately.

Instead, imagine making a change deep in a system, and simply telling Lattice's LMK™ to take over. No further thinking or keystrokes. LMK will rebuild your final product, however involved and complex, by doing just what is needed and no more.

How? You write a command file which expresses, bottom to top, all the elements comprising your system and all its dependencies: what gets compiled to make what object file using what options; what is built into libraries; what is linked into the final EXE file. Through the life of your system LMK keeps track of the last time every action was performed. Run LMK and, tracking each branch, it looks only for elements which changed later than a dependent element further along the branch, using date and time information in the file directory. Any source file newer than its object file, for example. Only those elements and their dependents are re-made. All other instructions are bypassed.

LMK does not care what programming language you use; it's not just for C. For that matter, LMK can apply to more than programming. It can be used for *any* set of tasks which can be accomplished through commands issued to the operating system.

Wherever your imagination leads you, LMK will find the shortest path to get the job done. Minimum time, minimum effort software.

Product Code: L2100	Our Price:
List Price: $195	$159

The TOOLSHED™

CODE SIFTER

Several expensive programs will now find the sluggish spots in your program, whereas our Code Sifter saves you plenty. It will divide a .COM or .EXE file of any size into 32 equal partitions. Or specify the partition boundaries with addresses, or with symbols, if your linker has produced a symbol map. Then tell Sifter to run your program. It samples your object file at precise intervals and counts how many times it finds the instruction pointer in each partition. Job done, it reports the number and percentage of hits in each, telling where best to optimize. You may be in for some surprises.

Product Code:	List Price:	Our Price:
N3100	$119	$99

CVUE

It's a text editor which does what a good editor should do plus undelete and full path name support in reading and writing files, and has features important to programmers: recognition of non-display characters, pattern searching, indenting and undenting of block structures.

Supports only in-memory text files, but with memory at today's prices, that could be 500k. Compact and fast. Runs in only 64k with no tedious overlays.

The real value: Even the binary version offers full customization of keyboard editing commands. Plus, take advantage of the source code option and make this editor truly your own.

Product:	List:	Our Price:
L2240 Binary	$ 75	$ 69
L2245 With Source	$250	$195

C-SPRITE

SOURCE ENHANCED

Lattice's Own Debugger for Lattice® C

We once called it a symbolic debugger, but Lattice advances now bring your source code on screen for your viewing pleasure. Hand this versatile companion to your compiler a .COM or .EXE file and C-Sprite™ will speak your language: your function names, your variable names, your data types, and the line numbers from your source code. At any breakpoint you can disassemble the object code and see source and assembler intermingled on screen.

If inclined, you can just as well get a close-up of machine addresses and machine-coded instructions to scrutinize what the compiler (or an assembler) contrived.

You already know how to converse with C-Sprite if you are familiar with Microsoft's Debug. Lattice began with that well-known command language, and added to it considerably: You can work with data in hex, as you might expect, but you can also differentiate between C's data types to cause the debugger to treat addresses as strings, long integers, etc., even pointers, both in display and entry.

C-Sprite can set breakpoints using symbols or addresses. You can submit clusters of commands to be executed at the breakpoints, or set commands that execute until a condition is met.

C-Sprite even has macros. Use your source code variable names in a macro to dump the contents of entire C structures, for example. And you can debug through one of the COM ports with a second terminal so as not to disturb your program's display screen. What's more, if you link with Plink86™, C-Sprite can even tackle overlays.

Product Code: L2300	Our Price:
List Price: $175	$145

CURSES

A Screen Management Interface to Swear By

Curses manages the screen of the IBM PC in the same fashion as the curses utility of Unix and similar operating systems. Use it to adapt programs which call Unix's curses functions for screen management, and need the equivalent library when moved to the PC for re-compilation. Or use it when creating software on the PC to assure that it is Unix compatible.

Curses is a library of eighty-four functions and macros which can keep any number of screen images in memory. A screen may be full or partial size, and any can be summoned to the physical screen at your programs's command.

Within a screen, Curses employs a vast function set to get characters, wrap lines, scroll, blank lines, highlight— virtually any conceivable tool needed to update the screen. The product supports color, and all four memory models. Its input functions give you control over whether to echo each character to a memory screen. In keeping with the terminal orientation of Unix curses, the physical screen is repainted (at high speed) only when your program calls a refresh function.

Writing screen management code leads to unspeakable snarls and expressions. Swear off! Let Curses clean up your language.

Code:	List Price:	Our Price:
L0850	$125	$100
L0860 with Source	$250	$200

EXPORT EXPERTISE

PC BRAND ships anywhere. We'll prepare the export documents and ship to you or your agent by air parcel post, air freight or courier. Phone or Telex your order for speed. To pay by credit card, please provide number, date of expiry, name and billing address of card. Or wire funds to PC BRAND, c/o Chemical Bank, 126 East 86th St., New York, N.Y. 10028, Account No: 034-016058. We will ship immediately and acknowledge by Telex if a number provided.

CRAFTSMAN™ QUALITY SOFTWARE FOR 'C' & OTHER PROGRAMMERS

C-TREE

B-Tree File Manager, Source Code, No Royalties!

A b-tree can be infested with bugs, so before buying one, ask its age. In a stand of saplings, this one is a real cequoia. C-tree™ has been around since 1979. (It became Digital Research's Access Manager™). That means seasoned, sturdy code which hasn't cracked under the weight of prolonged and widespread use.

C-tree comes in C source code, revealing all you've ever wanted to know about how b-trees are written. Provided you bind it into your binary application, you can re-distribute C-tree without royalties.

Not enough? Try this: C-tree's design splits nodes to allow any number of users to access an index file simultaneously even when updates are in progress. So multi-user configurations and adaptation to networks are possible. You must write record-locking routines, as they are compiler and operating system dependent, but the documentation shows how.

The latest version has new features: support of variable record length files; multiple key indexes in a single physical file.

Thanks to source code which does not deviate from the K&R standard, C-tree can travel. Binary equivalents have always meant finding a substitute file manager for migration to a different compiler, operating system, or computer; then changing all the function calls program-wide; then retesting. That's over. Tests in many environments prove that C-tree gives your application a ticket to anywhere.

C-tree permits any number of keys for a data file, supports duplicate keys, alphanumeric or numeric, etc., etc.—it's a comprehensive product with everything you'd expect. Intelligently designed, too, comprising both high level ISAM routines which minimize coding by handling all details of adding a record; as well as decomposed step-by-step functions you can access directly. Either way C-tree maintains optimal index structures which will find a record amongst a million tenbyte keys in no more than five disk seeks.

Product Code: F0660< — Our Price: $329
List Price: $395

TEXT TOOLBOX

New Expanded Version

Lattice has assembled a cluster of the most useful Unix™ text management tools into a single package. This new version neutralizes WordStar file idiosyncracies.

"Grep" looks for text patterns in any number of files. Want all occurrences of a global variable throughout a program system? Want to search all programs in a directory, down paths to other directories, or all files on a disk? Need to find all the function calls in an entire program system? Grep can do it with a powerful expression syntax that goes far beyond your text editor's search command.

"Splat" substitutes new text for matches found using grep's syntax. "Files" can copy or erase files down paths to the point of eliminating full directory structures.

"Diff" compares text files line for line. Its output is a precise list of instructions telling what to do to make two files the same, a list to give to "ed" to handle.

"Ed" is similar to the well-known Unix editor. It offers search and replace with grep syntax, block move, read-write, line numbering, append, insert, delete, etc. You can instruct "ed" to apply a file of commands to any number of target files, even complicated changes and text additions, such as output from "diff".

Product:	List:	Our Price:
L2200 Binary	$120	$99
L2205 With Source	$240	$195

PRE-C

Thorough "Lint"-like Analysis Now on the PC

Unix users have always had "lint" to thoroughly clean programs before they disappear into a compiler.

Pre-C™ looms larger than "lint". It finds problems your compiler won't. Problems that a debugger will have trouble figuring out. Even problems which will cause trouble with other compilers.

Pre-C finds all the syntactical tripwires that will blow out a compiler, sure, but it goes after subtler problems: code which will never be accessed, casts with suspect conversions, variables declared as external but never used, functions never called, obsolete usage (even C has changed), machine-dependent expressions which will inhibit portability.

But the big service of "lint" is this: Compilers work with one module at a time. They know nothing of other modules which only meet up at link time. Pre-C looks at *all segments of your program at once* and reports to you any inconsistencies of inter-module references: conflicting data type declarations; parameter lists in function calls which disagree with the functions themselves in number or data type; declarations of external functions which differ from their definition.

Pre-C uses the Unix System III compiler standard to guarantee maximum portability anywhere in the C world. There are then plentiful command line options to relax such rigor: forgive functions left incomplete during early coding, for example, or vouchsafe nested comments. Moreover, the output of each analysis can be filed for use with subsequent Pre-C runs, to avoid redundant work.

Use purchased binary libraries? Then store profiles of function names, their argument lists and their types, and Pre-C will thereafter insure that calls to them from any C program are letter perfect.

Big product! A miracle worker in speeding large system development. Needs 128k minimum; 192k recommended.

Product Code: P0590 — Our Price: $295
List Price: $395

TOOLWORKS C COMPILER

NEW!

Benchmarks Embarrass Biggies at Giveaway Price!

Borland move over. If you are looking for minimum cost tools for learning C or for getting straightforward jobs done, we are happy to end your search.

Take a peek at the prices below, then look at the benchmarks. We have here a bargain C compiler with some very respectable clockings against not only low price compilers, but even against big leaguers like Lattice. The timings will surprise you (will you look at floating point?).

Lattice it's not. Our Toolworks™ C doesn't have the full repertoire of features. But maybe you don't need them just yet. If the small memory model (64k for program and data each) will keep you busy for now, and you haven't yet found a reason for double precision math; if what you really need is to create C code and get it compiled, and you don't mind doing it for several hundred dollars less, then Toolworks C will be happy to generate remarkably fast and efficient compiled code for you.

To save the most, get the essential compiler—restricted to integer math values not exceeding ± 32k. Or order the Mathpak as well and receive a library capable of single precision floating point and long integers, as well as trigonometric, hyperbolic, logarithmic and power functions.

Toolworks C has a configuration program to change the compiler's default option settings, and a run-time profiler to uncork bottlenecks. It can generate either object code or assembler compatible with Microsoft's MASM, and gives you source code of its complete standard library, which includes access to DOS and BIOS interrupts.

Our compiler doesn't care what generation of DOS you are creating for. Toolworks C supports register variables, can use the 8087 math chip, even has an exec() function which will branch to another program or a DOS transient and return.

It's simply packaged. Nothing fancy. But our evaluators tell us that it is easier to use than most compilers, and no bugs were found. Not surprising. This compiler is not new. It started life under CP/M. Thousands of keybangers shaped the original to perfection's mold. We're so confident of reliability that if you are the first to document a bug, you can keep the compiler, and we'll refund your money.

Product TC001
Compiler & Runtime Library — $49.95
Product TM001
Mathpak for Floats & Longs — $29.95

30-DAY MONEY-BACK GUARANTEE

Even after using the program disks, you may return any product in this space not coded "<" (by requirement of its supplier) within 30 days for full refund of purchase price, provided it remains in our judgement in saleable condition.

BENCHMARK RESULTS Tests (key below): Best score indexed to 100.

	#1a	#1b	#2	#3	#4	#5	#6	#7	Compile Time	Compile Size
Toolworks	180	100	100	213	139	146/200	181/218	61k	162	135
Lattice		240	140	100	100	103/100	105/100	58k	205	121
DeSmet	670		113	107	100	100/130	100/113	60k	100	117
Ecosoft		160	115	116	110	101/130	105/143	fail	171	100

Test Key:
#1a: Floating point.
#1b: Same, using 8087 if capable.
#2: Function calls.
#3: Character counting.
#4: String copy using pointers.
#5: File copy using getc, putc; floppy/harddisk
#6: File creation, floppy/harddisk.
#7: Max. memory allocated using malloc().
Benchmark by DrasTech.

dBC *Lattice Library Maintains dBASE Compatible Files With the Power and Speed of C*

There are a lot of dBASE™ file users out there. Most of them just keep data bases and use dBASE's limited reporting facilities. They're not programmers, so they don't use the dBASE programming language. But they'd like more for their efforts, and that's a business opportunity.

dBC™ links C to dBASE. It creates and maintains files and their indexes which exactly replicate dBASE file design. So dBASE can read and update them. And the reverse. dBC can use any files created by dBASE Now C and dBASE can operate on the same data bases interchangeably.

That opens up the widespread culture of dBASE installations to exploitation by C programmers. Now you can replace the resident dBASE language with the speed of C. And you no longer have to write every line of code, as in the dBASE language, because now you've unlocked C's vast storehouse of off-the-shelf libraries and utilities.

dBC's functions parallel all dBASE's file handling commands, many decomposed to give closer control. The manual is backed by demo source files on disk.

Use dBC for custom work for clients, or use it on its own. It's a complete ISAM file manager for use with the Lattice C compiler whether or not dBASE will be used in tandem, supports all four memory models, and can have sixteen index and data files open.

Versions:	List:	Our Price:
L00II for dBASE II	$250	$225
LCCII with Source	$500	$450
L0III for dBASE III	$250	$225
LCIII with Source	$500	$450

Buy both II & III for Extra Discounts

TERMS AND CONDITIONS OF SALE

Licenses: Each price is for a license to use a product on a single computer and does not constitute its ownership. We will inquire for you about site licenses. Except for those coded "#", products may be used to create programs for distribution without royalty payments or additional licenses, provided said programs do not substantially replicate the products themselves.

Compatibility: PC BRAND's standard products are designed to operate with the IBM® PC, XT or AT under PC-DOS and require no more than 128K of RAM unless indicated. Non IBM machines using MS-DOS: Ask us about compatibility, but manufacturer knows best.

Returns: Defective parts will be replaced. Products coded "<" are rendered unreturnable if you open sealed envelopes containing diskettes. Otherwise, call for authorization to return a product for refund or credit.

Payment: We honor MasterCard, Visa, American Express (no surcharge), wired funds, checks in advance (in NY State, add sales tax), and will ship COD (U.S. only) for cash, money order, or certified check (no fee). Purchase orders accepted from larger corporations and institutions at our discretion if you agree to net 30 days plus 2% a month late penalty thereafter.

Shipping & Handling: U.S.: UPS Surface: 1st product $6, each add'l $3. UPS 2nd Day Air: 1st product $10, each add'l $4.50. UPS Next Day Air or Federal Express 1-2 Day Air: 1st product $18, each add'l $6. FedEx Next Day 10 AM: 1st product $28, each add'l $7. International: Charges vary by destination and carrier. $10 per shipping container for export forms. Air parcel post at your risk beyond collected insurable amount.

To Order Products or Catalog, Call Us at…

800-PC BRAND

That' (800) 722-7263. In NY State or Outside U.S. call (212) 410-4000.
PC Brand, P.O. Box 474, New York, N.Y. 10028
TELEX: 667962 (SOFT COMM NYK)

SAVE $5 Just tell us this code: J12

Prices, terms, and specifications subject to change without notice.

CIRCLE NO. 171 ON READER SERVICE CARD

Creating a Standard LISP

This preferred language for AI applications continues to grow and standardize; Gold Hill's GCLISP holds the promise of full implementation on microcomputers.

MARK BRIDGER and JOHN FRAMPTON

LISP implementations often have been handicapped by nonstandard dialects, unfriendly environments, and scanty documentation. Gold Hill's Golden Common LISP (GCLISP) breaks new ground in two ways: it brings the feel of a mainframe LISP environment to the PC, and it implements Common LISP, the language's new standard, to a degree. Programmers choosing GCLISP also will enjoy GMACS, the company's multibuffered full-screen editor, and the San Marco LISP Explorer, an engaging tutorial in LISP and artificial intelligence (AI) by San Marco Associates.

Although its adherence to the new standard is incomplete, GCLISP is a powerful language and Gold Hill promises even better features with version 2 (to be released in late 1985), including a truer implementation of Common LISP, a compiler, and a large memory option to take full advantage of the PC/AT. Clearly the company has undertaken an ambitious project. As it is, however, GCLISP is a fine educational package. The LISP Explorer offers some 60 lessons with text, graphics, and exercises in topics ranging from the basics to expert systems and natural language processing. The San Marco Inspector displays intricate function structures by tracing function calls graphically.

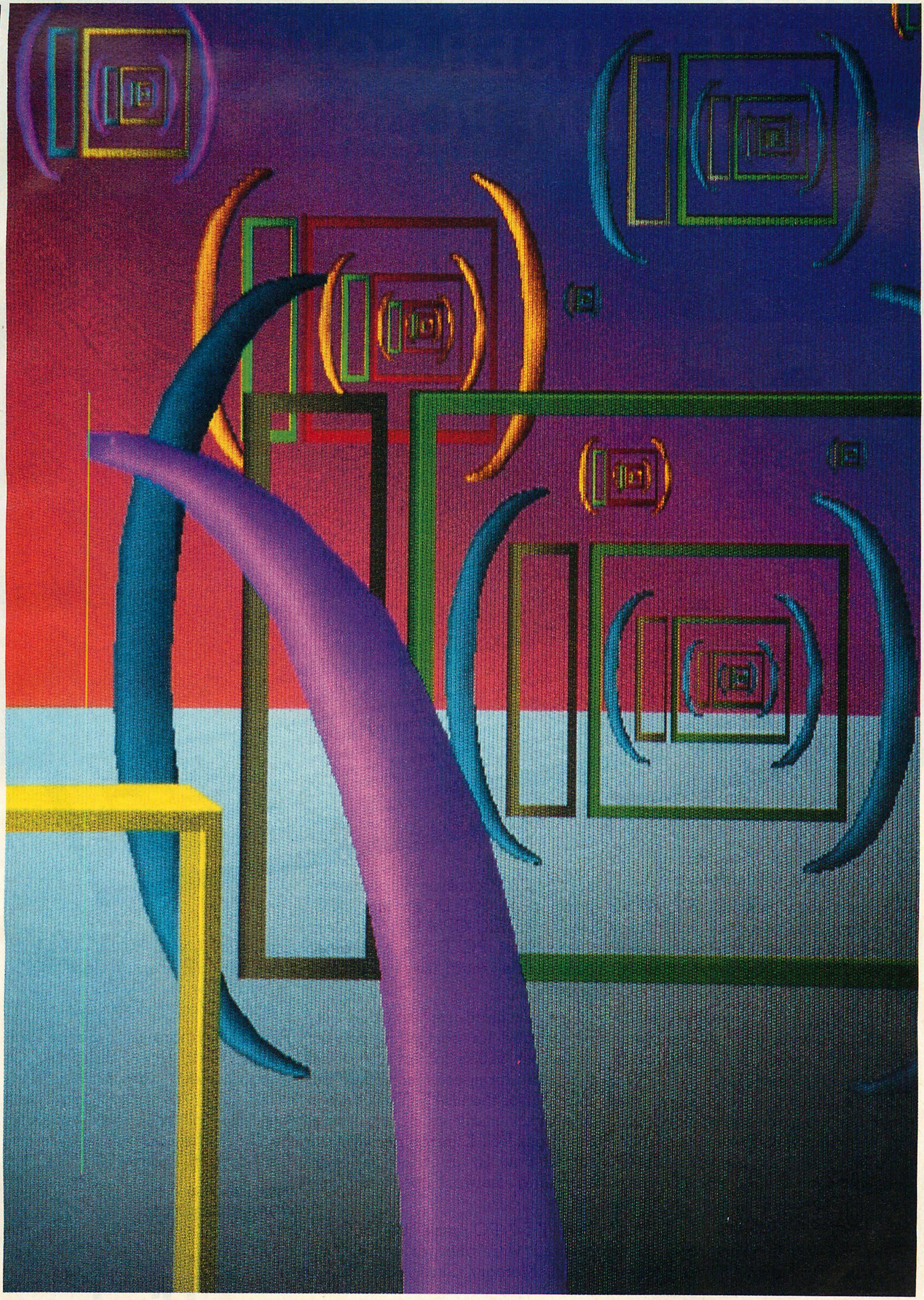
The experienced programmer will appreciate GCLISP's on-line documentation. Single keystroke commands list all symbols that have names containing a specified string, display documentation for a specified object, or list the parameter structure of a specified function. With the displays, a user is able to monitor an expanding LISP workspace by documenting his own functions.

A lot has happened to LISP since three implementations were reviewed in *PC Tech Journal* in April 1984 ("The PC Speaks LISP," William G. Wong, p. 112). In addition to GCLISP, recent releases of IQ LISP and muLISP (two of the three products reviewed) also provide partial support of Common LISP. This is welcome relief for programmers who have dealt with porting code among LISP's many dialects, the profusion of which is a sign of the strength of a language's basic concept.

Common LISP, the new standard, is the product of a national committee of LISP experts which assumed the task of standardizing and stabilizing the language and incorporating the major advances of the last 10 years. The committee's work is summarized in the excellent reference *Common LISP* by Guy L. Steele, Jr. (Digital Press, 1984); it is included in the GCLISP package.

LISP is by far the most important programming language for artificial intelligence; for more than 20 years its dialects have been the choice in most AI research and practical applications. However, perhaps because it is conceptually different from most languages, and in spite of recent increased attention fueled by the commercial success of some so-called intelligent systems written in LISP, it is largely unused.

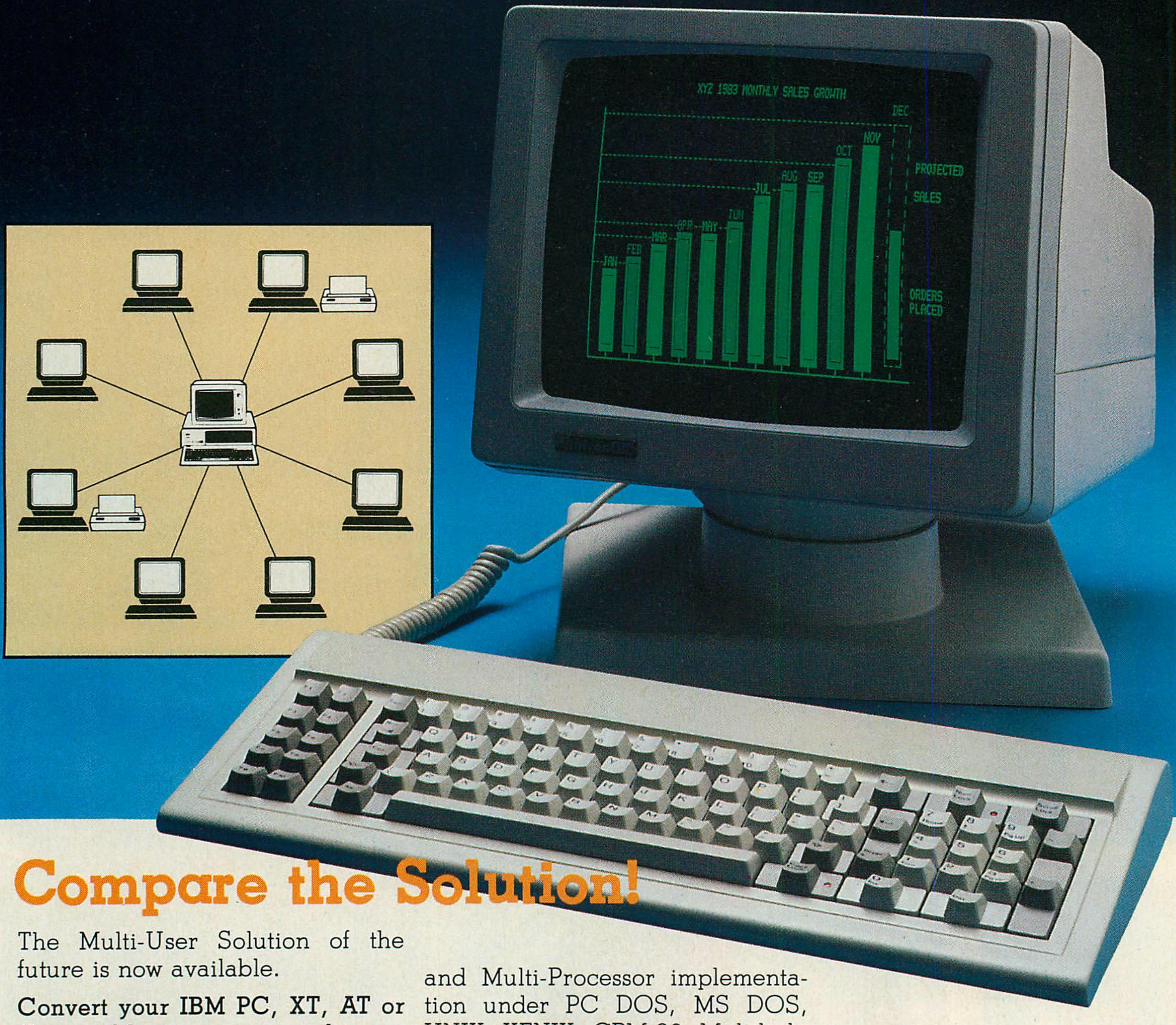
A discussion of GCLISP outside of AI circles, therefore, should begin with an update on LISP itself, including the effect that structured programming has had on the language. In addition, an explanation of the important features of Common LISP will help to put Gold



COMPUTER GRAPHICS • DOV JACOBSON

Kimtron

MULTI-USER SOLUTION for IBM PC, XT, AT



Compare the Solution!

The Multi-User Solution of the future is now available.

Convert your IBM PC, XT, AT or Compatibles to a true multi-user system while maintaining display, keyboard and software compatibility.

Since the KT-7/PC display is the same as your PC monochrome monitor, with its look-alike keyboard, operators will feel they're using an IBM PC and can also use the same software manual. Kimtron's multi-user solution includes file and record locking, shared data access, and communication between users. It is **the** intelligent alternative.

The KT-7/PC supports Time Sharing, Enhanced Time Sharing and Multi-Processor implementation under PC DOS, MS DOS, UNIX, XENIX, CPM 86, Multilink, Concurrent PC DOS, and other compatible multi-user operating systems.

Kimtron's multi-user solution may be tailored for cost effectiveness; as low as $1095 for an additional user, and for speeds more than ten times faster than LAN. You can add one or as many as 31 additional users per PC. Kimtron delivers the future now by allowing an ever-widening network of multi-user PC's.

The KT-7/PC may be complemented with one (or more) I/O Card, Memory Card, 8086 Speed Enhancer Card, 80286 AT Card, 8088 Multi-Processor Card, 80186 Speed Enhancer Card, 68000 Card, and related software.

For more information about Kimtron's Multi-User Solution, or general video data terminals for other mini or micro multi-user systems, call your local computer dealers, distributors or Kimtron Corporation Today!

(408) 286-8790
TWX: 910-338-0237

1705 Junction Court
Building #160
San Jose, CA 95112

Kimtron

NOTE: IBM PC, XT, AT, PC DOS, MS DOS, UNIX, XENIX, CPM 86, Multi-Link, Concurrent PC DOS are registered trademarks of IBM Corporation, Microsoft Corp., Bell Labs., Digital Research Inc., Software Link Inc. respectively.

CIRCLE NO. 188 ON READER SERVICE CARD

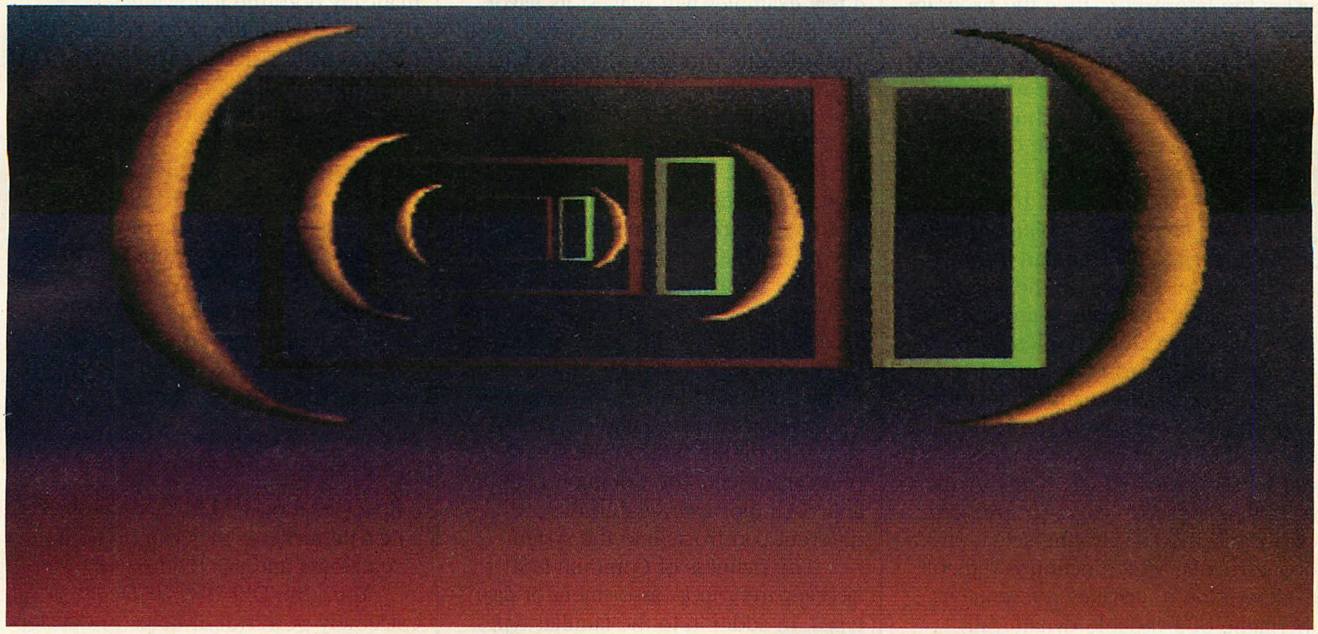
See us at COMDEX Fall '85 Booth #4566

Hill's implementation of the new standard into perspective.

LISP manipulates extremely diverse and complex data objects using its fundamental structured data type—a list (its name comes from list processing). Because LISP is highly recursive, lists of lists of lists (and so on) become available to the programmer. An extremely broad range of objects can be represented easily by these structures in such a way that they actually reflect the intrinsic structure of the object. It is principally for this reason that LISP is so important for symbolic processing. LISP implementations provide the wide variety of constructs necessary to manipulating list structures.

The fundamental LISP tool for data manipulation is the function, which is analogous to a Pascal function, but which often exhibits side effects that can make the returned value irrelevant. A LISP program typically consists of the definitions of a number of functions that can call each other, predefined functions (GCLISP provides a few hundred of these), or, recursively, themselves. A large LISP program is built of hundreds of such definitions.

The execution of the program is accomplished by the evaluation of one of these functions, analogous to the main body of a Pascal or C program. Very significantly, the definition of a LISP function itself has a list structure, so that a LISP program can modify itself, easily manipulate LISP code as input or output, store the definitions of functions in data structures for access by other functions, and so on. In short, LISP can treat programs as data.



LISP pays much more attention to the environment created for program development than do most languages. Special editors designed to manipulate list structures and that can be accessed directly within the LISP environment have been developed. Extensive facilities for incremental program development—for tracing or stepping through program execution or trapping runtime errors and suspending execution—have been implemented.

Two major influences have shaped recent LISP developments. First, as AI has begun to move out of the laboratory, efforts have increased to improve its efficiency to make it competitive with languages such as C and Pascal in execution speed. Historically, LISP had been willing to pay the price for its flexibility and generality in a slow execution. Second, LISP has been influenced by the move toward structured programming and the demand for textual clarity of source code. Fortunately, these two influences have dovetailed nicely to produce major improvements.

Although the core of LISP has remained stable, the language now thrives in many dialects. Early LISP dialects had only one loop construct (the PROG function) in addition to the mapping functions. GCLISP has an additional five, all much more structured than PROG; they are LOOP, DO, DO*, DOLIST, and DOTIMES. This is typical of modern LISPs, although the constructs and syntax details vary from dialect to dialect.

LISP have always been implemented as an interpreted language; its facilities for interactive debugging and incremental program development have been based upon this. However, compilers capable of improving execution speed by a factor of 10 also are available. The drive to produce efficient programs has forced modern LISPs to pay increasing attention to compilation and to language features that permit the production of high-quality object code.

Historically, most dialects of LISP have exhibited a peculiar feature: the interpreter and compiler have implemented somewhat different languages. In particular, the binding of variables (the association of values with names) has been treated as *dynamic* by the interpreter, but as *static* by the compiler.

The main concern is how free variables within a function receive values. A free variable is one that occurs in the function definition but is neither a formal parameter nor a local variable. In a language with dynamic binding, a free variable gets its value at the time the function is called, looking into the calling environment for the most recent assignment of a value.

In static binding, the bindings of free variables are determined when the function is defined by the environment of the function definition. This environment is determined statically, from the textual structure of the program. For this reason static binding is referred to as *lexical scoping.* LISP compilers make this the default inasmuch as compilation is simplified if the manner in which variables are to be computed at runtime can be determined at compile time from the textual structure of the program. LISP compilers commonly establish dynamic binding for a variable only if a declaration to that effect is made.

While dynamic binding offers definite advantages, it is weak in bringing to bear LISP's potential for treating functions as data. The problem is that the references of free variables are not determined until a function is called. It is sometimes impossible to guarantee that free variables of a function passed as a parameter to another function will be properly referenced.

The split personality between compiled and interpreted LISP has led to a two-step process of creating compiled LISP programs. A program is developed using the interpreter, then the program is examined and the changes necessary to make it compile are made. Often this produces LISP programs in which the interpreter runs correctly, but that cannot be modified to compile correctly without extensive rewriting.

Modern LISPs require that interpreters and compilers implement the same language. In particular, they choose lexical scoping as the default for both the interpreter and compiler. Dynamic binding can be requested explicitly using the appropriate construct.

Aside from eliminating difficulties in program development, static binding offers other significant advantages: It can be implemented efficiently by the compiler. It greatly increases program clarity and modularity (the actual effect of a particular LISP function is difficult to determine if it makes reference to the name of a variable dynamically bound in some distant part of the program). Perhaps most importantly, lexical scoping allows a significantly fuller use of functions as data (as arguments to or values of other functions).

TABLE 1: *The SETF Macro*

ACCESS FORM	SAMPLE UPDATE USING THE SETF MACRO
X	(SETF X "three blind mice")
(CAR X)	(SETF (CAR X) 5)
(TURTLE-COLOR TURTLE1)	(SETF (TURTLE-COLOR TURTLE1) 'red)
(AREF X 3)	(SETF (AREF X 3) 4.5)

Common LISP's SETF macro allows destructive replacement within any structure. It is a clear improvement over the REPLACA!s and SELECT-ELEMENTs of other LISPs.

COMMON LISP

This definition is an attempt to summarize the development of modern LISP and to create a de facto standard. It was conceived over the period from 1981 to 1984 (primarily through electronic mail) by a committee of researchers from the leading centers of AI research across the United States. The pressure for a standard arose as a combination of the recent flourishing of LISP dialects and the desire for portability of the increasing amount of LISP code being written for both commercial and military applications.

Common LISP is a large language. An earlier failed attempt at standardizing (Standard LISP) demonstrated that a proposal must offer a very rich, full-featured language; if it does not, dialects that rely heavily on constructs outside the proposed standard will develop quickly. The richness of Common LISP cannot be given full treatment here, but a few examples will suggest its flavor.

Some LISP implementations have had only one conditional function, COND, which is (roughly) structured like a multiple alternative nest of IF..THEN statements in Pascal. The major difference is that no boolean expressions following the first one that evaluates to true are themselves evaluated. Common LISP, on the other hand, also provides IF (a standard if..then..else alternative), UNLESS, WHEN, CASE, and TYPECASE. Clearly, the logic of each of these constructs could be handled easily by the COND function—TYPECASE, for example, makes a choice of tasks to carry out on the basis of the data type of an argument. The Common LISP approach, however, is to build the different constructs into the system.

The richness of Common LISP is also apparent in its assortment of data types. In addition to the familiar types, such things as hash tables, read tables, path names, and streams are provided. Further, functions for manipulating objects of each type are provided; for example, seven different functions are available for manipulating hash tables. Types are divided into subtypes. Common LISP specifies the following subtypes of number: integers (including both fixnums and bignums), ratio (of two integers), rational (integer or ratio), floating point (further subdivided into short, single, double, and long varieties), and complex.

One data type is new to LISP. Common LISP and GCLISP provide for user-defined *structures,* which are much like records in Pascal and other languages, but with a useful twist. The following code defines turtle as a new data object with four named components:

```
(defstruct turtle
  x-position
  y-position
  friends
  color)
```

To this point it is just like Pascal. But Common LISP takes the process further. The evaluation of the code above automatically does the following:

- Creates access functions TURTLE-X-POSITION, TURTLE-Y-POSITION, TURTLE-HEADING, and TURTLE-COLOR, each of which is a function of one argument (of type turtle) that can retrieve the components of a turtle.
- Creates a function TURTLE-P which can test if an object is a turtle.
- Creates a constructor function MAKE-TURTLE, the evaluation of which creates a data object of the desired type that allows for the components of the object to be initialized using a key-word syntax as follows:

```
(setf my-turtle (make-turtle :color 'green
  :x-position 12 :y-position 37))
```

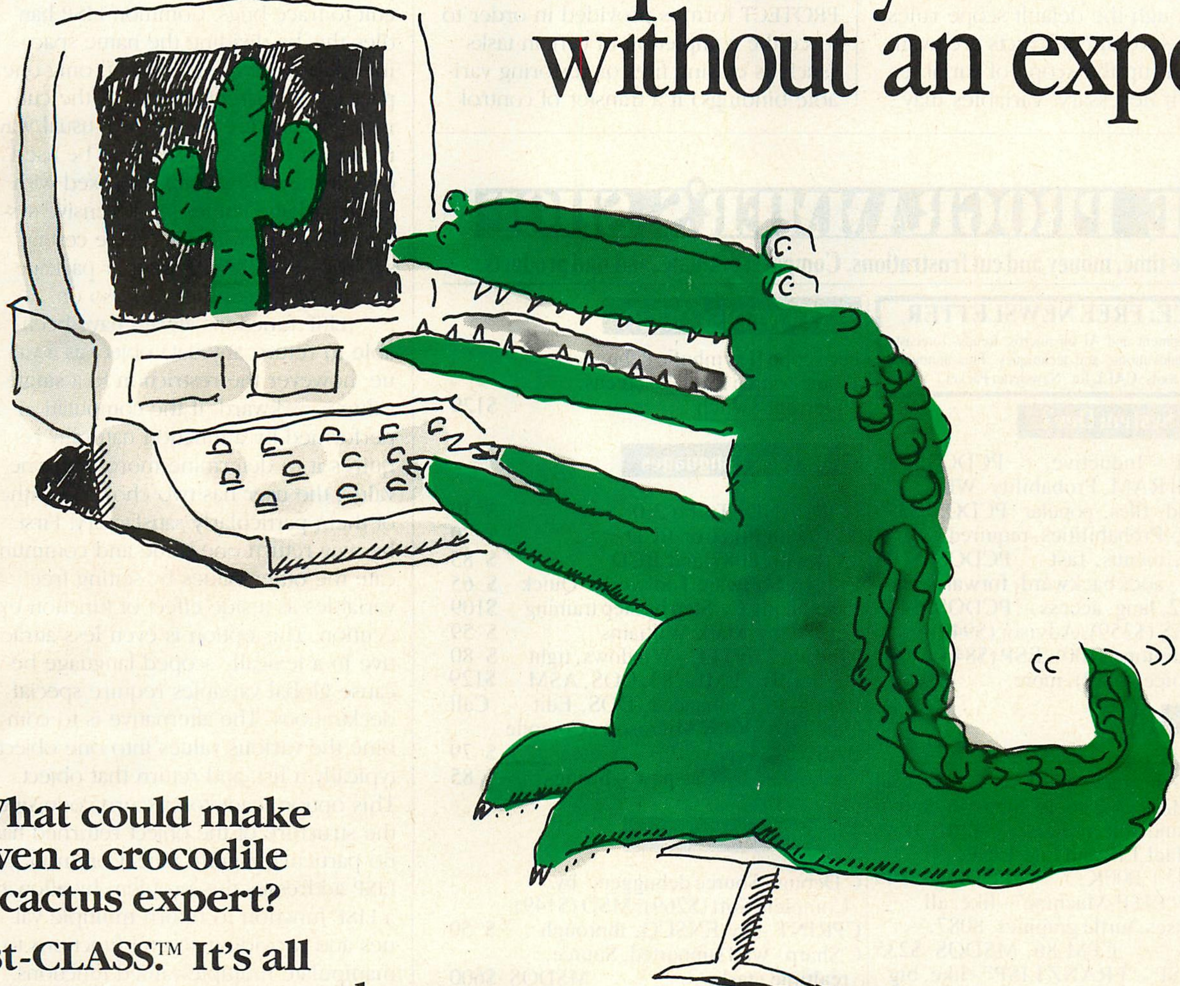
- Specifies a printed format for objects of type turtle so they can be both read from and printed to a text file.

Thus, the DEFSTRUCT facility is extremely powerful. It helps in developing efficient code and, consequently, very readable programs.

LISP dialects have had a bewildering collection of functions for which the roles are simply to access or update values associated with particular variables. For example, the function CAR accesses the first element of a list; RPLACA replaces the first element with some other given value. In some dialects SELECT-ELEMENT accesses array elements and REPLACE-ELEMENT replaces an array element with a given value. The list of such pairs is quite long.

Common LISP and GCLISP bring some order to this chaos with *generalized variables.* The concept of a variable named by a symbol is generalized to that of a storage location identified by an access form. In place of the simple assignment of a value to a variable, a value must somehow be assigned to a generalized variable. This is provided by the SETF macro, which uses the relevant access function to specify the appropriate location to update. Table 1 provides some illustrations. (AREF, for example, is the Common LISP equivalent of SELECT-ELEMENT.)

Build an expert system without an expert.



What could make even a crocodile a cactus expert?

1st-CLASS.™ It's all the expert you need.

With 1st-CLASS, you can build an expert system of any size or complexity without a knowledge engineer. In fact, you may not even need an expert.

Makes its own rules. Just identify the problem, then give it examples. It generates rules based on what it teaches itself from the examples. It can diagnose a problem, automate complex procedures, and even make predictions from confusing data.

Speaks your language. 1st-CLASS develops a query system for the end-user based on your own words. And not only does it give the end-user an answer, it also tells how the decision was reached and gives probabilities of success.

All this in a program that's as easy to use as a spreadsheet.

Talented, resourceful. 1st-CLASS works with both logical decisions and numbers. It can use them right from your spreadsheet or database. It can also cue any other hardware or software for integrating, for example, slides into the program. Once it's made a decision, it can tell another program about it.

The kicker: only $250. 1st-CLASS expert system operates on IBM-PC and compatible computers, DOS 2.0 or higher. For only $250, you get the rule generator, the query system, interface programs, the manual, example systems, and updates for one year. Or, for $50, get 1st-CLASS Intro, a fully functional version for smaller problems.

Call 617-879-9650 to order, or for free literature.

Or write:
Programs in Motion Inc.
10 Sycamore Road
Wayland, MA 01778

Although SETF renders the traditional update functions of LISP superfluous, a few (RPLACA, SETQ, and SET) have been retained for continuity.

FUNCTIONS AND SCOPING

Common LISP is a lexically scoped language: no distinction exists between the way a Common LISP interpreter and compiler attach meaning to a LISP program. Although the default scope rules are lexical, several constructs are available to open up the scope of variables and labels if necessary. Variables may be declared to have dynamic scope by invoking a suitable form. At the highest level, this results in a global variable.

The new standard also provides the CATCH and THROW constructs. CATCH establishes a dynamically scoped label that can be the target for transfer of control by the subsequent evaluation of a THROW form (this makes a long distance GOTO possible). The UNWIND-PROTECT form is provided in order to force the completion of certain tasks (such as closing files or restoring variable bindings) if a transfer of control such as that effected by a THROW, or an error, takes place.

Common LISP solves the problem of name conflict. In large LISP programs, which can contain several distinct modules (such as editors and operating system support), a problem develops in keeping the variable names used in the different modules from colliding and producing strange and difficult-to-trace bugs. Common LISP handles this by dividing the name space into *packages*. At any instant, only one package is current. Names in the current package are used in the usual way; names in other packages can be used only if they are suitably prefixed with their package names. An extensive system also is provided to make certain symbols available in another package handle name conflicts, and so on.

LISP functions always have been able to return any data object as a value; however the restriction to a single value is awkward. If the computation performed by a function naturally requires it to determine more than one value, the user has two choices, neither of them particularly satisfactory. First, he may return one value and communicate the other values by setting free variables as a side effect of function execution. This option is even less attractive in a lexically scoped language because global variables require special declarations. The alternative is to combine the various values into one object, typically a list, and return that object. This option leads to obscure code since the structure of the object returned has no particular significance. Common LISP addresses this problem by allowing a LISP function to return multiple values and providing several functions to manipulate multiple-valued functions.

The syntax of function definition in Common LISP also allows functions to be defined by the programmer with optional parameters that are given prescribed default values if the function is called with no corresponding value specified. Certain parameters can be specified by *key words* rather than by position on the argument list.

The question as to whether Common LISP will make it as a standard may receive a military answer. Indications have the U.S. Department of Defense, through its Advanced Research Projects Agency (ARPA) and Strategic Computing Project (a major funder of AI research), pushing it strongly. The DOD would prefer that its various supported research efforts be able to share LISP code. Its quality has won Common LISP a valuable endorsement.

PERFORMANCE PACKAGE

Blaise Computing Inc. introduces the PERFORMANCE PACKAGE™ for Turbo Pascal programmers.

TURBO PASCAL

Turbo ASYNCH™

With Turbo ASYNCH, you can be in constant touch with the world without ever leaving the console. Rapid transit at its best. Turbo ASYNCH is designed to let you incorporate asynchronous communication capabilities into your Turbo Pascal application programs, and it will drive any asynchronous device via the RS232 ports, like printers, plotters, modems or even other computers. Turbo ASYNCH is fast, accurate and lives up to its specs. Features include...

◆ Initialization of the COM ports allowing you to set all transmission options. ◆ Interrupt processing. ◆ Data transfer between circular queues and communications ports. ◆ Simultaneous buffered input and output to both COM ports. ◆ Transmission speeds up to 9600 Baud. ◆ Input and output queues as large as you wish. ◆ XON/XOFF protocol.

The underlying functions of Turbo ASYNCH are carefully crafted in assembler for efficiency, and drive the UART and programmable interrupt controller chips directly. These functions, installed as a runtime resident system, require just 3.2K bytes. The interface to the assembler routines is written in Turbo Pascal.

The Turbo Pascal PERFORMANCE PACKAGE™ is for the serious Turbo Pascal programmer who wants quality tools to develop applications. Every system comes with a comprehensive User Reference Manual, all source code and useful sample programs. They require an IBM PC or compatible, utilizing MS-DOS version 2.0 or later. There are no royalties for incorporating PERFORMANCE PACKAGE functions into your applications.

Turbo POWER TOOLS and Turbo ASYNCH sell for $99.95 each, and they may be ordered directly from **Blaise Computing, Inc. TO ORDER, call (415) 540-5441**

NEW!

Turbo POWER TOOLS™

Turbo POWER TOOLS is a sleek new series of procedures designed specifically to complement Turbo Pascal on IBM and compatible computers. Every component in Turbo POWER TOOLS is precision engineered to give you fluid and responsive handling, with all the options you need packed into its clean lines. High performance and full instrumentation, including...

◆ Extensive string handling to complement the powerful Turbo Pascal functions. ◆ Screen support and window management, giving you fast direct access to the screen without using BIOS calls. ◆ Access to BIOS and DOS services, including DOS 3.0 and the IBM AT. ◆ Full program control by allowing you to execute any other program from within your Turbo Pascal application. ◆ Interrupt service routines written entirely in Turbo Pascal. Assembly code is not required even to service hardware interrupts like the keyboard or clock.

Using Turbo POWER TOOLS, you can now "filter" the keyboard or even DOS, and create your own "sidekickable" applications.

YES, send me the Best for the Best! Enclosed is ________ for
☐ Turbo ASYNCH ☐ Turbo POWER TOOLS. (CA residents add 6½% Sales Tax. All orders add $6.00 for shipping.)

Name: ________ **Phone:** ________
Shipping Address: ________ **State:** ________ **Zip:** ________
City: ________ **Exp. Date:** ________
VISA or MC #: ________

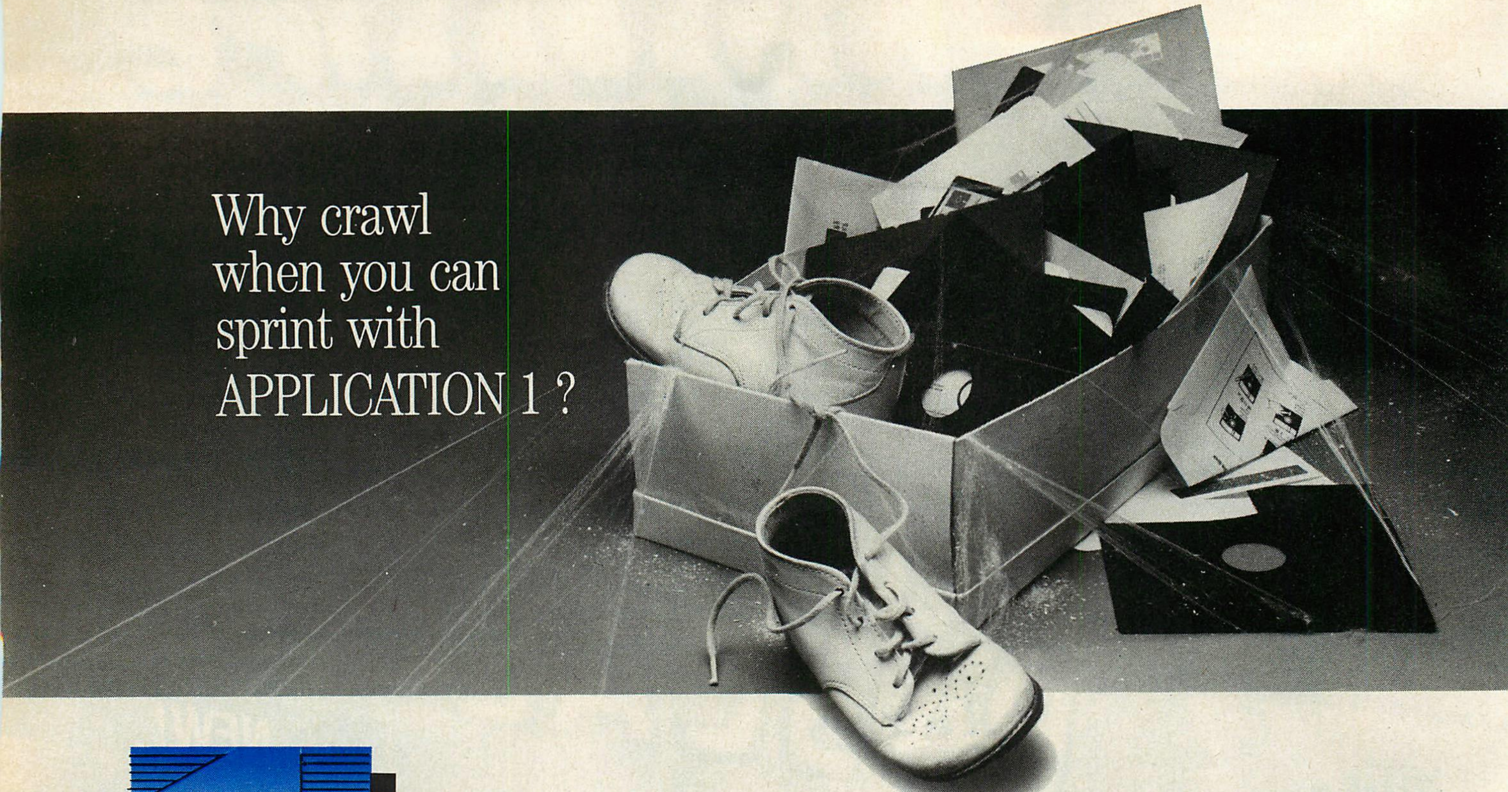
Turbo Pascal is a trademark of Borland International. Turbo POWER TOOLS, Turbo ASYNCH and PERFORMANCE PACKAGE are trademarks of Blaise Computing Inc. IBM is a registered trademark of Internationl Business Machines Corporation. MS-DOS is a trademark of Microsoft Corporation.

BLAISE COMPUTING INC.

BLAISE *watch us!*

◆ 2034 BLAKE STREET
◆ BERKELEY, CA 94704
◆ (415) 540-5441

CIRCLE NO. 114 ON READER SERVICE CARD



Assembler speed in IBM Basic and Microsoft "C". Mainframe power for your IBM PC. Create network applications.* Choice of file or record locking.

Today's programmers want more than development tools and Data Base Managers that merely crawl when they need to sprint. APPLICATION 1's rich assembler routines combined with IBM Basic or Microsoft "C", sprints during the creation and execution mode of your software...and that means up to 6 times faster and more!

Its remarkable efficiency is a result of this unique file system. Achieve unperceivable performance differences from files containing 300 to as many as 80,000 records or more. Design screens in minutes with Genscr, our full screen editor. And most important, APPLICATION 1 opens up the door to file sharing in update mode. Now.

Conserve precious disk space with variable length record and key indexes. With unlimited files, fields, records and 8087/80287 support, APPLICATION 1 has it all.

You can grow with us because we bring no limitation to your imagination and future requirements.

We've prepared a "Demonstration Kit" for Application 1. This includes complete documentation and demo disk. Order yours today for $49.95 (US) which is refundable on the purchase of the $595.00 (US) full version. Contact Logiquest at 514-397-9631.

LOGIQUEST

Logiquest Inc.
555 Dorchester West, Suite 1616
Montreal, Quebec, Canada H2Z 1B1
1-514-397-9631

Required:
PC DOS 2.1 and up
PC DOS 3.1 and up for networking
256k Ram.
IBM PC, XT, AT, and compatible
IBM Network Cards presently supported*

CIRCLE NO. 202 ON READER SERVICE CARD

LESS THAN A SUBSET

A full implementation of Common LISP on mainframe computers takes between 1MB and 2.5MB of RAM, not including editors and other modules. Any implementation on the PC must make a judicious selection of features since only a subset of the full language can be included. In its porting, GCLISP is something less than a subset.

Nearly every implementation of a language provides some features over and above the standard. In a true subset the user might expect a correct program in the subset (that has been written *without* the additional features) will remain a correct program in the full language, carrying out the same computation, except perhaps for features such as file handling that are necessarily implementation-dependent.

GCLISP does not yet implement lexical scoping of variables—a feature central to Common LISP. Gold Hill has said that version 2 will have lexical scoping, but until then, GCLISP cannot be considered a true subset of the new standard even though it does share many of its features.

GCLISP does not implement lexical scoping of variables —a feature that is central to Common LISP; Gold Hill says that version 2 will.

Most of the syntax of function definition is available, including multiple-value functions. However, Common LISP allows a function to return no values. This is helpful to a user who wants to employ a function purely for its side effect or who wants to avoid the return of an irrelevant value. A GCLISP function must return at least one value. In addition, specification of function parameters by key words rather than by position is not available for user-defined functions although many built-in functions take key-word arguments.

GCLISP's generalized variable facility provides the core of the full Common LISP facility, including its most useful features. A major departure is its nonstandard method for constructing new generalized variables. (Gold Hill's manual says this will likely change.)

Virtually all of Common LISP's control structures are implemented (functions that control the flow of program execution, such as conditionals, iteration constructs, CATCH, THROW, UNWIND-PROTECT, BLOCK, RETURN-FROM, RETURN); a few esoteric structures are omitted. This is important because control structures provide the backbone for program construction. Programmers learning LISP with the GCLISP product do not have to worry about an exposure to inferior methods.

GCLISP's DEFSTRUCT facility has all of the most important features of the Common LISP facility. It comes as a separate file that is loaded automatically when the DEFSTRUCT macro (a function in Common LISP) is invoked. Several other GCLISP facilities are loaded similarly; this saves precious RAM in situations where they are not needed.

In line with its overall approach of providing services at the most general level, Common LISP provides a number of functions designed to work on sequences, defined to be either a list or a vector (a one-dimensional array). This is one area in which GCLISP cuts corners. Only functions SUBSEQ(uence)

and LENGTH are available for application to either lists or vectors. The SORT function (which takes two arguments, a sequence, and a predicate to establish order) is listed in the GCLISP reference manual, but is not implemented. This is unfortunate as SORT can be quite useful. Nor is MERGE implemented.

GCLISP does not provide the range of number subtypes that are specified by Common LISP; complex and rational numbers are not available. In the standard, integers are a subtype of rational and come in two varieties, fixnum and bignum. In GCLISP, only fixnum (as a subtype of number) is available; it offers integers in the usual PC range of −32767 to +32767.

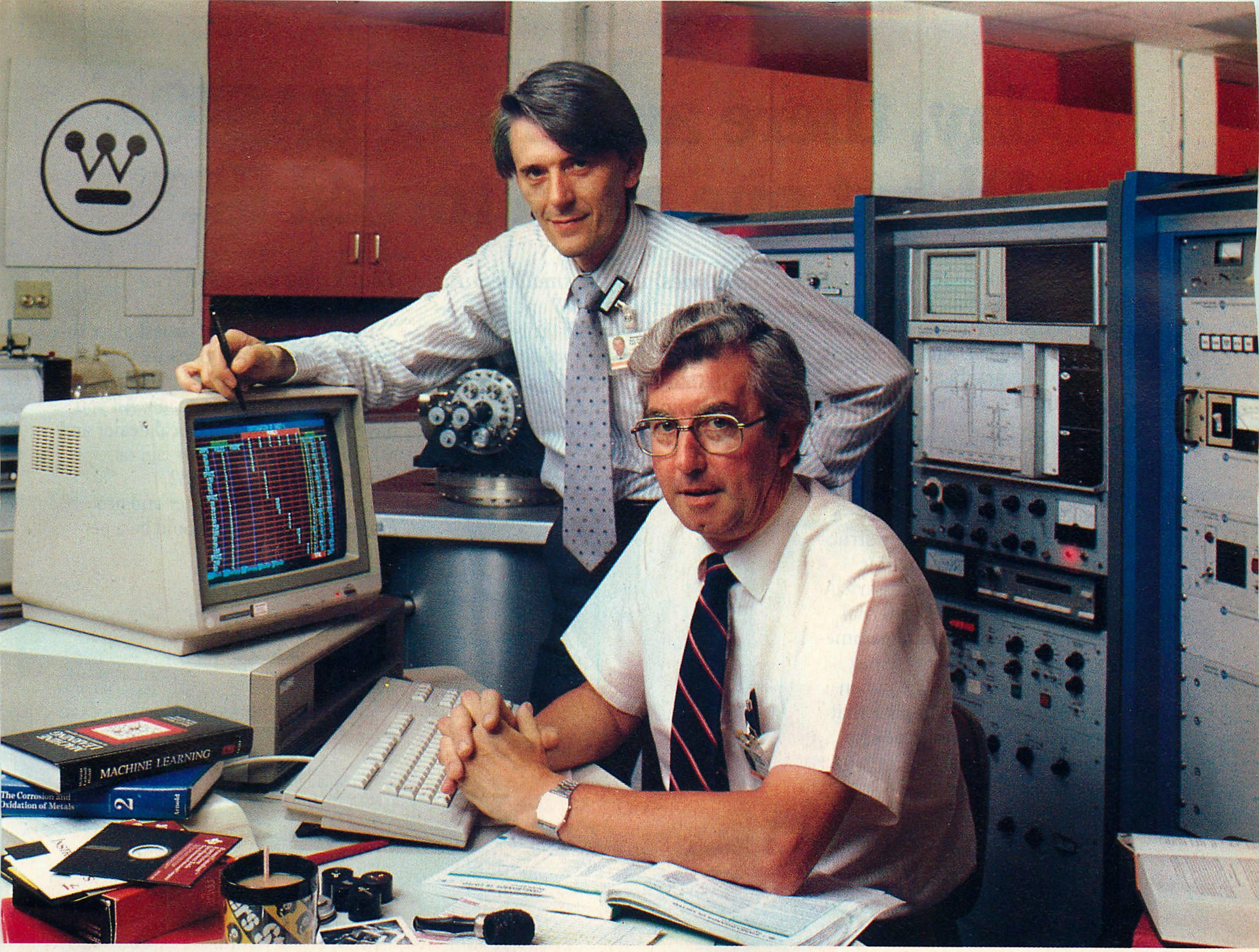
Both single and double floating-point subtypes of the type float are provided; they correspond to the Intel 8087 short and long real number formats. The 8087 support is restricted to arithmetic operations: none of the transcendental functions (sin, cos, exp, etc.) is available, either via the 8087 or via software. (They appear in the reference manual but are disclaimed in the release notes.) This limits GCLISP's usefulness in some respects, but will not hinder it for most AI applications. Gold Hill plans a fix in version 2.

Also promised in version 2 is a full implementation of the Common LISP package system; it is only partial now. The editor, for example, currently runs in the same name space as the user. This could have disastrous results from name conflicts. One likely collision point is the function DIV (not a GCLISP function), the definition for which is located in the GMACS file EDKEYS.LSP. Users defining the function DIV might be surprised at the consequences. The editor uses other names (TOCK for example) that easily could be redefined by an unsuspecting user. When Gold Hill fully implements Common LISP's package system, the editor will have its own name space and these problems will disappear.

Like Common LISP, GCLISP file handling is heavily oriented toward files of characters. Files of unsigned bytes are available, however, for handling binary data. The file handling functions work quite well. Furnishing the appropriate optional parameters to Common LISP read functions makes it possible to trap "read past end of file" errors and return any specified value—a convenient feature. Its I/O functions do not actually apply to files but to streams that are connected to files. Streams have possibilities other than connections to files. GCLISP supports windowing via an extensive syntax for the creation and manipulation of window streams. The creation of character string streams is also possible; these can be manipulated with the I/O functions, a convenient syntax for building editors.

The very convenient DRIBBLE function can be used to echo all terminal input and output to a specified file as a record of an interactive session.

GCLISP provides good system services. It can be interrupted and control transferred to DOS for execution of commands, provided sufficient RAM has been configured. Normally, a user reserves only enough RAM for DOS to provide basic services; control is returned to GCLISP by a DOS EXIT. Interrupt service is provided directly by a multiple-valued function. The parameters are loaded into the registers, the interrupt generated, and the new register contents returned as multiple values. Low-level functions that peek and poke RAM and others that access the I/O ports also are provided. Arithmetic functions for unsigned integers are included, a very useful supplement to the



Westinghouse brought in TI's Personal Consultant™ because these experts don't have time to waste.

Personal Consultant is the expert systems development software that's turning the time they spend answering routine questions into research and development.

Dr. Neil Pessall and Dr. Jan Schreurs, research scientists at the Westinghouse Research and Development Center, had a problem. How could they make hundreds of man-years worth of practical knowledge at the Center readily available to Westinghouse and its customers, without pulling the scientists away from their research to answer routine questions?

Their solution was Personal Consultant, a new expert systems development software tool from Texas Instruments for TI or IBM® personal computers.

Designed to run on 512K-Class Personal Computers, Personal Consultant provided Westinghouse an eminently affordable new tool for the development of expert systems. Pessall and Schreurs were especially impressed with its ability to aid experts in developing the knowledge bases and prototyping programs which even first-time computer users could consult for expert advice.

"With Personal Consultant software," says Schreurs, "we found it far easier to teach our experts how to develop a system than teach a programmer to be an expert." In fact, the software's rapid expert system prototyping and simple debugging features enabled the two scientists to demonstrate three different expert systems in as many weeks. Schreurs produced one for training new salespeople and a second to guide repairs and maintenance for the Center's X-ray defractometer. Pessall's makes material recommendations for tubes and support plates in steam generators.

"Our development of expert systems isn't aimed at replacing experts," says Dr. Pessall. "Its purpose is to give them the time to extend their expertise while the expert systems handle consultations on well-established knowledge and methodology."

To put Personal Consultant from TI to work for your business, call (800) 527-3500, in Canada call (416) 884-9181. Because an expert's time is too valuable to waste.

TEXAS INSTRUMENTS

Creating useful products and services for you.

CIRCLE NO. 170 ON READER SERVICE CARD

low-level functions provided and much needed for system-level work.

Given the low-level functions available, it is unfortunate that Gold Hill did not document the internal structure of GCLISP for programmers with special needs. No mechanism or information is provided for linking to foreign code. This is another area scheduled for revision in the next version, with significant structural documentation as well as a standard mechanism for linking to Lattice C procedures.

In general, the GCLISP documentation is undistinguished; its manual does not provide a thorough discussion of GCLISP specifics. For example, it fails to point out that the GCLISP function STRING-APPEND is not a Common LISP function (Common LISP uses the sequence function CONCATENATE which is unavailable in GCLISP). Most users will look instead to *Common LISP* as their primary reference.

More weakness is exhibited by GCLISP documentation in its discussion of stack groups, a GCLISP extension of Common LISP. This provision offers user-defined evaluation stacks that can be interrupted and then resumed, just as with the main evaluation stack, to enable the user to build subprocesses—*coroutines*—parallel because all computations are in process at the same time, although only one is active at any time.

In general, the GCLISP documentation is undistinguished; the manual does not provide a thorough discussion of GCLISP specifics. Most users will want to look instead to Common LISP *as their primary reference.*

An interesting theory, but only LISP experts could use the manual's scant hints to take advantage of GCLISP's support for stack groups.

In addition to its written documentation, GCLISP provides on-line help, which will speak well of it for some. Whether or not this is acceptable is a matter of personal preference; in either case, a product of this caliber should provide the user with better manuals.

The excellent on-line help facilities should make GCLISP particularly good for teaching purposes where it is not uncommon for the user to work without a reference manual. Three of its very useful functions can be invoked by a simple Alt-key command: APROPOS lists all symbols with names that contain a specified string. DOC displays useful information for a specified function, variable, or type. LAMBDA-LIST displays the parameter structure for a specified function. The information for these functions is stored on a very full double-sided diskette which is accessed when the function is invoked.

The two-disk San Marco LISP Explorer, included in the GCLISP package, offers an introduction to programming in LISP. The format is 57 "slide shows" on topics such as list constructors, branching with COND, using structures, and translating with macros. Explorer runs from the LISP environment. After viewing one of the slide shows, the user can write and run examples from that topic, and each show includes a number of exercises.

The Explorer is by no means a complete introduction to LISP or even to GCLISP. It does not replace user's guides, instruction manuals, or textbooks, and is of little use to the experienced LISP programmer; however, it does provide a good hands-on introduction to LISP and computer algorithms. Often the temptation in learning a new language is to read manuals long past the time when actual programming should begin. The Explorer helps to avert this by providing immediate interaction with LISP.

LISP users have come to expect powerful debugging facilities; indeed, this is one of the language's major features. GCLISP lives up to these expectations and includes the full set of LISP debugging features: breakpoints (via the function BREAK), display of the list of forms that are awaiting evaluation (via BACKTRACE), function execution tracing (via TRACE), and the macro STEP for stepping through the evaluation of any form that uses specified keyboard commands to control the course of evaluation.

The GMACS full-screen editor is a joy to use; it uses multiple buffers, with the option of two windows. More importantly, it provides many features specifically designed for editing LISP code. A left parenthesis that matches a right parenthesis immediately to the left of the cursor blinks on and off. Many users will appreciate this feature. LISP forms can be evaluated without leaving the editor, making possible rapid-fire testing and redefinition. Users can toggle back and forth between LISP and the editor with simple key strokes. In addition, a single key command will produce the expansion of a macro the instant its definition is complete. As expected, this editor includes move, delete, and insert commands that work on list structure rather than line structure.

The editor is written in LISP, so that the advanced user could easily customize it—given good documentation. The documentation, however, falls

The Explorer is not a complete introduction to LISP or even to GCLISP, but it provides a good hands-on orientation to LISP and computer algorithms.

short; Gold Hill does not even explain how to change the editor key commands. A little digging in the files will turn up the function DEF-KEY, which can be used for this task when the command tree structure for the keys has been discovered. The key commands carry little redeeming mnemonic value; changing them is worth the effort.

THE PRESENT AND FUTURE

Some natural language programs and expert systems have been written with PC GCLISP, but the memory demands of LISP AI applications indicate that an IBM PC will rarely be adequate to the task (the PC/AT is a different story, and its capabilities are discussed below).

This does not mean that LISP is out of place on the PC. This implementation can play a significant instructional role. In addition, the highly modular nature of LISP makes it possible to use a microcomputer to do experimental and developmental work on a program being developed for a larger computer.

mbp COBOL for your IBM/PC

Now, more convenient... 4 (exclusive) ways.

Now, the mbp COBOL Compiler offers unrivaled convenience to go with its unmatched performance.

Here are the convenience features you've wished for:

1) an enhanced Screen Management System with program-controlled video attributes and color; 2) support for PATH & sub-directories; 3) DOS command execution from within a COBOL program; 4) 'permanent' DEFAULT modification.

The new mbp Compiler has them all! And they're exclusives: you get them *only* with mbp.

GIBSON MIX Benchmark Results
Calculated S-Profile
(Representative COBOL statement mix)
Execution time ratio

mbp* COBOL	Level II* COBOL	R-M* COBOL	Microsoft* COBOL
1.00	4.08	5.98	6.18

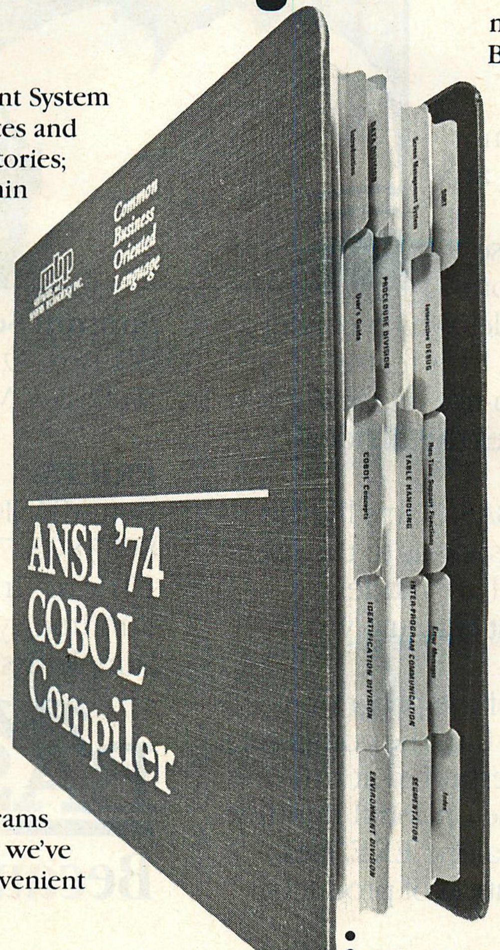
*128K system with hard disk required IBM PC & AT are IBM TMs. Netware is a Novell, Inc. TM. Level II is a Micro Focus TM. R-M is a Ryan-McFarland TM. Microsoft is a Microsoft TM.

Plus, it's 4 times faster.

Because the mbp COBOL Compiler generates native machine language object code, it executes programs *at least* 4 times faster (see chart). Now, we've made that performance even more convenient to use.

The complete COBOL.

An Interactive Symbolic Debug Package included standard; Multi-keyed ISAM structure; SORT & CHAIN; GSA certification to ANSI '74 Level II; IBM/PC-AT and TI Professional compatibility; with mbp, you get it all. Optional: IBM-PC Network and Novell interfaces.



mbp COBOL: the choice of professionals.

It's no surprise more and more companies like Bechtel, Bank of America, Chase, Citicorp, Connecticut Mutual, Hughes Aircraft, McDonnell-Douglass, and Price-Waterhouse choose mbp COBOL.

Make it your choice, too. Just send the coupon, or call, for complete information. Today.

Visit us at COMDEX, Booth 4658

mbp COBOL. $1000

Please send complete mbp COBOL information to:

NAME ____________

COMPANY ____________

ADDRESS ____________

CITY/STATE/ZIP ____________

PHONE ____________

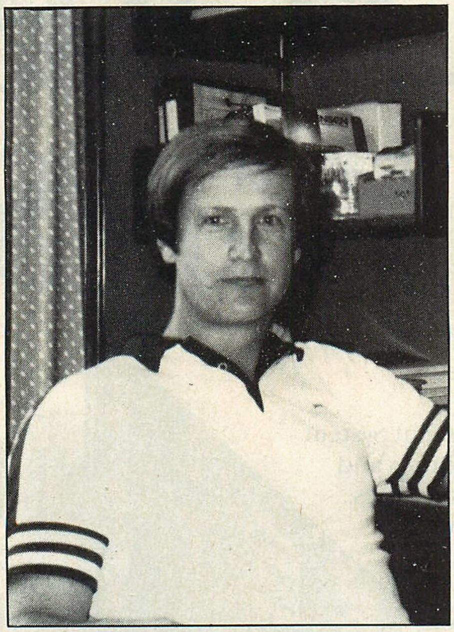
mbp Software & Systems Technology, Inc.
1131 Harbor Bay Parkway, Suite 260
Alameda, CA 94501
Phone 415/769-5333

mbp®

"I Program In BetterBASIC And I Recommend It."

"I've been programming for about eight or nine years in BASIC and I'm really impressed with BetterBASIC. It's got expanded memory, windowing and extensibility. I like the cleverness of the way it's been constructed using workspaces, modules, functions and procedures. Right now I'm working on three custom business applications and all three are being written in Better-BASIC because of all the things it can do. I program in BetterBASIC and I recommend it."

Bob Briggs - Software Consultant - North Easton, MA.



640K Now you can use the full memory of your PC to develop large programs.
STRUCTURED Create well organized programs using procedures and functions that are easily identified and understood and completely reusable in future programs.
MODULAR Use procedures and functions grouped together to form "library modules" which are then available to you or anyone else for future use.
EXTENSIBLE Create your own BetterBASIC modules which contain BetterBASIC extensions. This feature coupled with the easy-to-use Assembly Language support, makes this an ideal OEM language.
INTERACTIVE BetterBASIC acts like an interpreter because it responds to the users' commands in an immediate mode. However each statement is actually compiled as it is entered.
COMPILED Each line of the program is compiled as it is entered into the computer's memory rather than interpreted at runtime.
RUNTIME SYSTEM The optional Runtime System generates stand alone EXE. files allowing for the distribution of products written in BetterBASIC with no royalties.
SUPPORTS Windows, Graphics, DOS and BIOS ROM calls, Chaining, Overlays, Local and Global Variables, Recursion ... and more.
BetterBASIC Runs on IBM PC, XT, AT and all IBM-compatibles. Ask your local dealer for BetterBASIC or call 1-800-225-5800 in Canada call 416-469-5244. Also available for the Tandy 1000, 1200, AND 2000 AT Tandy/Radio Shack stores.

Summit Software Technology, Inc.™
P.O. Box 99, Babson Park
Wellesley, MA 02157

See Us At
COMDEX
Lifeboat's Booth
#1392

PRICES:

BetterBASIC	$199
8087/80287 Math Module	$99
BTrieve™ Interface	$99
Runtime System	$250
Sample Disk with Tutorial	$10

Better BASIC.

Because It's The Best.

MasterCard, Visa, Checks, Money Order, C.O.D. accepted and P.O. on approval.

(If you're using BetterBASIC and would like to be featured in one of our ads, please write to the Director of Advertising at Summit.)

CIRCLE NO. 196 ON READER SERVICE CARD

TABLE 2: *Benchmark Results*

	GCLISP on PC/AT	GCLISP on PC/XT	IQLISP on PC/XT
ITERATION			
PROG	.17	.47	.82
DO	.11	.28	
DOTIMES	.006	.02	
ITERATIVE LIST BUILDING	.06	.25	1.26
RECURSIVE LIST BUILDING	.22	.58	.96
FLOATING-POINT ARITHMETIC			
Without an 8087	—	2.42	7.97
With an 8087	—	.27	1.42

Times are in seconds.

The benchmarks time iterations and recursion. GCLISP trades compactness for speed; it uses four-byte pointers, IQLISP uses packed three-byte pointers.

Adherence to the Common LISP standard, or to a powerful subset of the standard, makes these uses much more attractive. If code developed in GCLISP on an IBM PC will run as is on a larger machine, the PC's usefulness in developmental work is clear.

GCLISP, therefore, shows much promise in giving microcomputers a role in AI work. However, some of its present weaknesses hamper its ability to fulfill this promise; the most important of these is its failure, thus far, to adopt lexical scoping. Development work performed in a language that attaches different meaning to code than the language in which the code is eventually intended is futile. This weakness also makes GCLISP much less attractive for instructional use than it might otherwise be. Presumably, instructors will want to teach Common LISP. The whole question of lexical binding and its influence on the form of control structures and the flow of control is central, to say nothing of the entirely different style of programming that develops in a lexically scoped LISP as opposed to a dynamically scoped LISP.

The product's lack of transcendental 8087 support is not significant overall, but its failure to implement Common LISP's package structure is a problem. Larger applications simply are unable to live happily in an undivided name space.

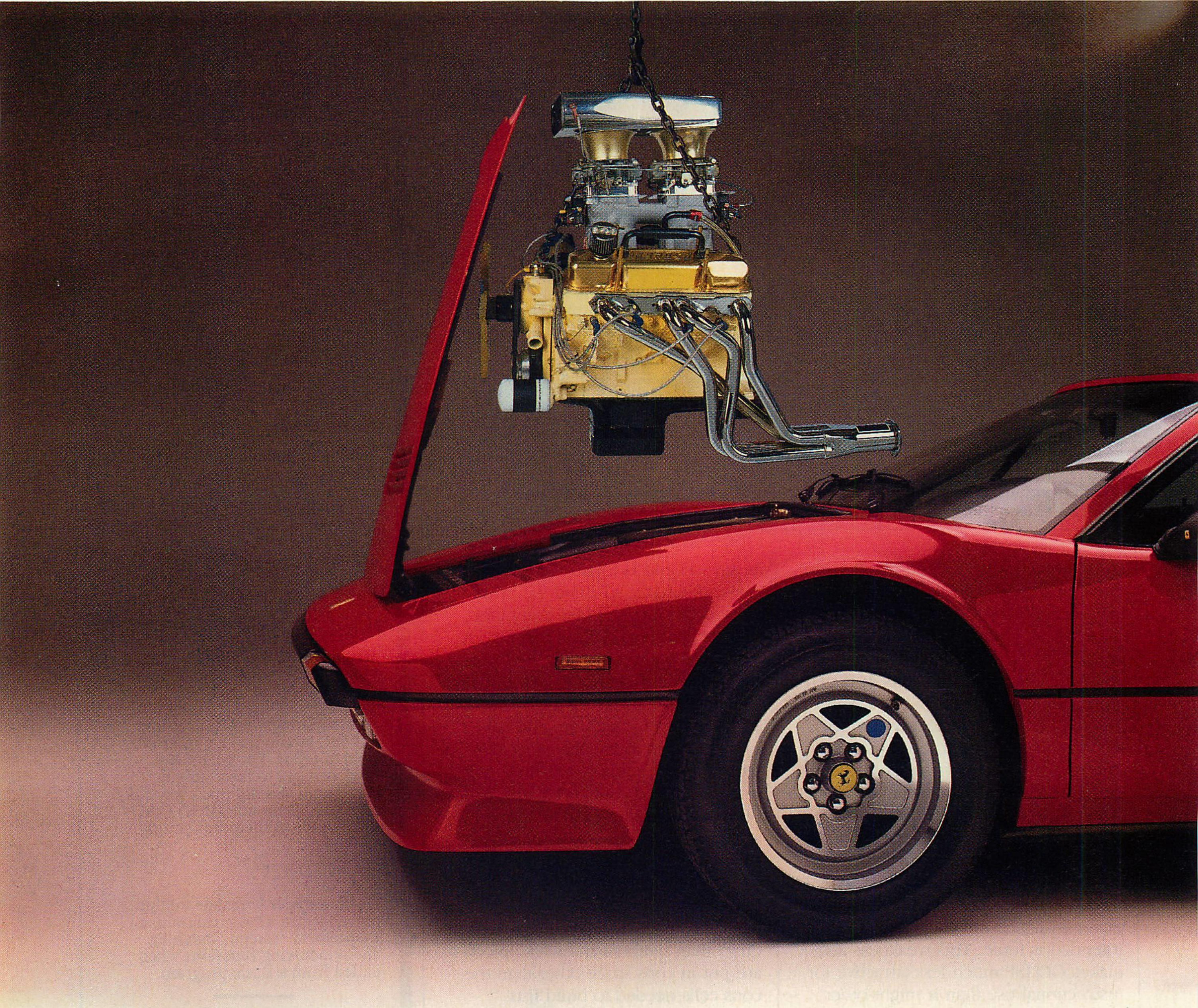
The performance of GCLISP on the PC/AT is better. The AT has the memory capacity and power for serious LISP applications programs. The company has announced an LM (large memory) superset of GCLISP (version 2) for the AT that will address a full 16MB of physical memory. (It is scheduled for release in late 1985; the projected price is $695.) Gold Hill also is working on networking PCs, ATs, and the Symbolics, Inc. special-purpose LISP computer.

GCLISP uses a lot of memory: A full 480KB is needed with the editor loaded. With the cost of RAM plummeting, it does not make sense to spend a lot of money for software and fail to bring a PC up to its maximum memory. GCLISP was tested on a PC with 640KB of memory and configured with 28KB on reserve for DOS; the memory was allocated on a 2 to 1 basis between *atom space* and *cons space.* (The ratio can be set in a start-up file.) Because of the meager documentation, it is uncertain what objects are put in atom space other than symbols and what objects are put in cons space other than the cons cells needed to build lists.

Each list entry requires one cons cell (nine bytes in this implementation) for the necessary list structure, in addition to whatever memory the entry object itself requires. This configuration resulted in a 135KB free atom space and 30,903 free cons cells when GCLISP was loaded without the editor. When the editor was loaded, the free atom space dropped to 104KB and the count of free cons cells dropped to 13,854.

The speed of execution was tested in several ways. (See table 2 for benchmark results and listing 1 for the source code used in all of the tests.) The size of the GCLISP stack is such that recursive function calls can be nested only to a depth of approximately 230. In order to make the comparison between recursion and iteration easy, all of the iterative loops used 200 iterations. Some execution times for IQ LISP are provided for comparison.

Iteration. An empty loop was run using PROG, DO, and DOTIMES. PROG is a



AT™ Pfantasies for your PC or XT.™

Want better speed and memory on your PC or XT without buying an AT?

You've got it!

Phoenix's new Pfaster™286 co-processor board turns your PC or XT into a high-speed engine 60 percent faster than an AT. Three times faster than an XT. It even supports PCs with third-party hard disks. But that's only the beginning.

You can handle spreadsheets and programs you never thought possible. Set up RAM disks in both 8088 and 80286 memory for linkage editor overlays or super-high-speed disk caching. All with Pfaster286's 1mb of standard RAM, expandable to 2mb, and dual-mode design.

You can develop 8086/186/286 software on your XT faster. Execute 95 percent of the application packages that run on the AT, excluding those that require fancy I/O capabilities your PC or XT hardware just isn't designed to handle. Queue multi-copy, multi-format print jobs for spooling. Or, switch to native 8088 mode to handle hardware-dependent programs and back again without rebooting. All with Pfaster286's compatible ROM software. And, Pfaster286 does the job unintrusively! No motherboard to exchange. No wires to solder. No chips to pull. Just plug it into a standard card slot, and type the magic word, "PFAST."

If you really didn't want an AT in the first place, just what it could do for you, call or write: Phoenix Computer Products Corp., 320 Norwood Park South, Norwood, MA 02062; (800) 344-7200. In Massachusetts, 617-762-5030.

Programmers' Pfantasies™ by Phoenix

XT and AT are trademarks of International Business Machines Corporation. Pfaster286 and Programmers' Pfantasies are trademarks of Phoenix Computer Products Corporation.
For the Ferrari aficionado: yes, we know this is a rear engine car. We are showing the addition of a second engine to symbolize how Pfaster can be added to your PC or XT to increase performance.

CIRCLE NO. 169 ON READER SERVICE CARD

SEE US AT FALL COMDEX

general control structure that uses labels and goto to build loops. Historically it was an important construct, however it is now giving way to more structured methods available in modern LISP. DO is a structured alternative to PROG that provides automatic looping, built-in structure for modifying local variables after each pass through the loop, and built-in structure to test for loop termination. DOTIMES is the Common LISP structured iterative construct for use when the number of iterations is known in advance.

List building. A list of length 200 was built using DOTIMES.

Recursion. A list of length 200 was built recursively.

Floating-point operations. This test ran 200 floating-point divisions, both with and without an 8087.

GCLISP is a powerful implementation of LISP and an excellent teaching tool. The San Marco LISP Explorer and Inspector provide for an enjoyable and productive learning experience. Gold Hill's version of the EMACS editor is a state-of-the-art programmer's tool. GCLISP's only major drawback is its departure from Common LISP in the area of lexical scoping. Of course, this and many other features and revisions are promised in version 2, including a compiler. Version 1.01 is a good product; version 2 should be excellent.

The cost to upgrade from version 1.01 will be $100. Users who plan to purchase the compiler need not be concerned about the update charge: version 2 will be supplied free with the compiler, which will sell for $495. Users interested in GCLISP would do well to wait for version 2.

Golden Common LISP 1.01: $495
Gold Hill Computers
163 Harvard Street
Cambridge, MA 02139
617/492-2071
CIRCLE 361 ON READER SERVICE CARD

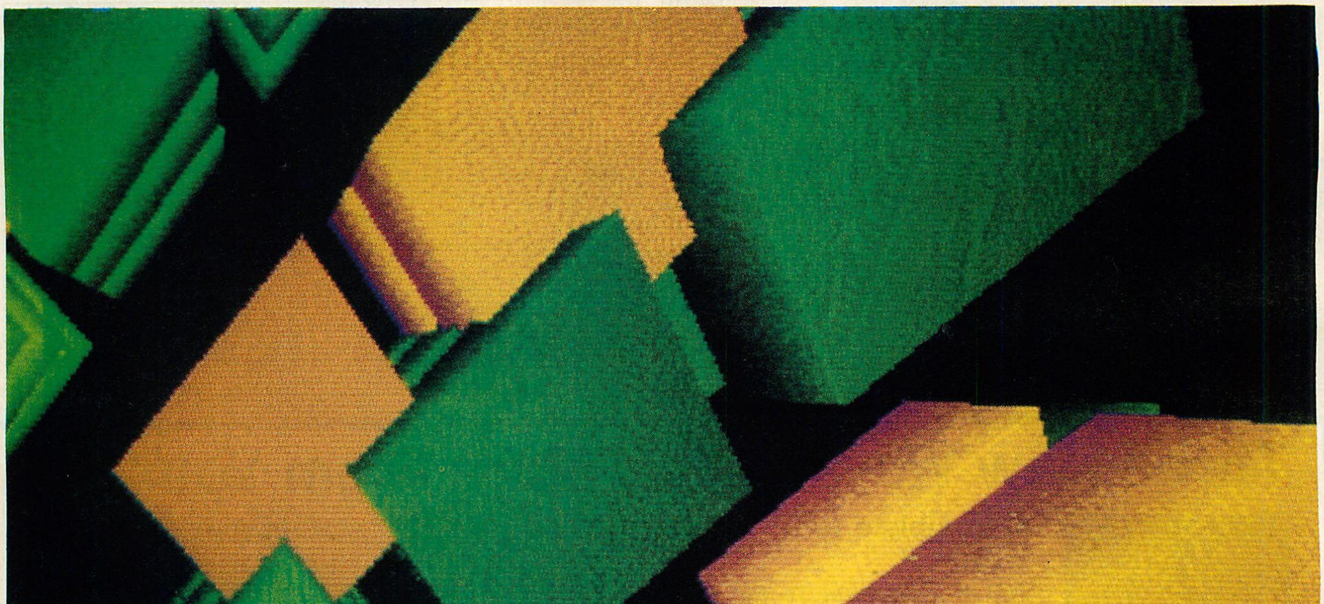
Mark Bridger and John Frampton are professors of mathematics at Northeastern University in Boston. Professor Bridger is an algebraist with a strong interest in computer-assisted instruction. Professor Frampton does research in artificial intelligence, especially in the area of natural languages.

LISTING 1: BENCHMRK.LSP

```
(defun empty-loop-prog (count)
  (prog ((n count))
    loop
    (when (zerop n) (return nil))
    (setf n (1- n))
    (go loop) ))

(defun empty-loop-do (count)
  (do
    ((n count (1- n)))
    ((zerop n) nil) ))

(defun empty-loop-dotimes (count)
    (dotimes (k count nil)) )

(defun makelist-iterative (count)
  (let ((result nil))
    (dotimes (k count result)
      (setf result (cons nil result)) )))

(defun makelist-recursive (count)
  (if (zerop count)
    nil
    (cons nil (makelist-recursive (1- count))) ))

(defun float-test (count)
  (dotimes (k count)
     (/ 1.0 3.0) ))
```

DATA MANAGERS AS DEVELOPMENT TOOLS

A Data Manager with a UNIX Heritage



COMPUTER GRAPHIC • DOV JACOBSON

Relational Database Systems has released a DOS version of its INFORMIX-SQL package, grounded in the UNIX operating system.

AUGIE HANSEN

In spite of statements to the contrary, many sound and useful business applications are available for UNIX systems. As a data manager for UNIX-based systems, INFORMIX has been around since the turn of the decade, serving the needs of the multiuser community quite well. Relational Database Systems (RDS) has recently released INFORMIX-SQL in a single-user version for DOS as well, and in early September it announced a networked version (however that product was not available for testing in time for this review). The UNIX version of INFORMIX has a runtime package available, but no runtime facilities are yet available for DOS.

Well over 200 third-party products based on RDS database products are now in circulation. INFORMIX-SQL is the basis of many commercial applications that range from banking and financial analysis to inventory control to retail trade management.

The primary language of INFORMIX-SQL is RDSQL, which is an extended version of IBM's original SQL (structured query language). The RDSQL language adds the ability to load and unload delimited database files, to change databases while operating, to alter the names and structures of tables in a database, as well as to use audit trails in order to provide transaction records and help in restoring corrupted database files and indexes.

An IBM PC or equivalent machine with 512KB of RAM and running DOS 2.0 or later is required for INFORMIX-SQL. Although the product can run on a floppy-disk-only system, a hard disk is recommended, especially for applications program developers. A little more than 900KB of disk space is required to hold the programs, message files, and miscellaneous files that comprise INFORMIX-SQL.

The package consists of separate modules for preparing and running reports, forms, and interactive queries.

INFORMIX-SQL is a relational data management system offering considerable flexibility in the way data items are retrieved and converted into useful information. It uses the classic database terminology in which a database is a collection of tables and related indexes, with each table containing rows and columns. In some other systems, tables correspond to database files, rows correspond to records, and columns correspond to fields. INFORMIX-SQL keeps the data files (tables), indexes, and some system files for a given database in a subordinate directory. The system files retain the bookkeeping and database structure information.

> ***A little more than 900KB of memory is required to hold all of the programs, message files, and miscellaneous files that comprise the INFORMIX-SQL package.***

INFORMIX-SQL is built around another RDS product, C-ISAM, a proprietary file-access system available in both UNIX and DOS versions. C-ISAM uses a B+tree indexing structure, permitting an unlimited number of secondary indexes, and index compression is used to conserve precious disk space.

Unlike many programs that constrain users to tight and often arbitrary limits on the sizes and number of elements that can be created and managed, INFORMIX-SQL is unfettered. The number of tables in a database, columns per database, and rows per table are unlimited by the program; available secondary storage sets the practical upper bounds. Likewise, the sizes and number of columns in a row are restricted only by the amount of addressable memory in the machine.

Row size is limited to a generous 32KB, and composite indexes may be created using up to eight columns (120 bytes maximum). Fifteen tables can be open simultaneously.

A variety of data types is available to the programmer. Characters and strings are specified as type CHAR(size). The size of a string is part of its type. The DATE type is used to declare columns that can hold date information presented in a range of formats. Special format control statements control the way dates are entered and displayed.

Whole numbers may be stored as INTEGER or SMALLINT types. INTEGER is comparable to C's signed **long** (32 bits in this implementation) and SMALLINT is a 16-bit signed quantity. INFORMIX has no unsigned data type.

Real numbers may be represented in several forms. A FLOAT type is the same as a C **double** type, and SMALLFLOAT corresponds to C's **float**. These have approximately 14 and 7 bits of precision, respectively. A variation on FLOAT is the DECIMAL type, with definable scale and precision. Values in the range of 0.1 * 10 ^ 126 to −0.1 * 10 ^ −128 with 32 significant digits may be stored. A special case of the DECIMAL type is MONEY, which always has two digits to the right of the decimal point.

The SERIAL data type is a unique identifier assigned to each record. It

may be assigned an initial value by the user, but then is handled entirely within INFORMIX-SQL. Each table has one SERIAL column, which is most often used for sequential row numbering.

INFORMIX-SQL does not provide for the use of arrays; nor does it have a logical (Boolean) type and memo/note types. Logical values may be stored in an integer or character variable for which the range of values is restricted to two. In the sample application used for this article, numeric 1 and 0 were inserted, but *T* and *F* characters also would have been permissible. (For a complete description of the sample database application used to test all of the data management products reviewed by *PC Tech Journal*, refer to "Sample Application Specifications," August 1985, p. 48; also available for downloading from PCTECHline, 301/576-PCTJ).

A large character string may hold textual input up to the maximum size specified, but a text space is not free to grow dynamically as text is added. The lack of array and memo types causes problems in handling the sample case of an indeterminate number of authors per article. Most of the alternatives, such as duplicating rows for an article for each different author, produce most undesirable solutions. Assigning storage for *N* authors in every row is wasteful because most articles are written by only one person.

The control system for INFORMIX-SQL is the main ISQL module. Other modules in the system may be run either on their own or from the ISQL menus. Statements provided for maintaining the database and its tables include the following: ALTER TABLE, CHECK TABLE, RECOVER TABLE, and REPAIR TABLE. In conjunction with these tools, a useful audit facility tracks modifications made to a table.

After a database table has been created and populated with data, it still can be modified with relative ease. The command to restructure a database table is ALTER TABLE, which has options to allow adding columns (ADD.. [BEFORE]), deleting columns (DROP), and modifying the data types of existing columns (MODIFY). Because the type of a character column includes its size, changing the size of a character field is considered a type modification.

INFORMIX-SQL performs automatic data conversions whenever columns are changed from one type to another, but common sense always must prevail. Impossible conversions, such as changing character strings that contain embedded punctuation marks into one of the numeric types, will undoubtedly fail, as will an attempt to convert high-value floating-point data into small integers.

APPLICATIONS PROGRAMMING

One of the benefits of a UNIX-derived product is its full support of operating-system path names for files. The recommended location for INFORMIX-SQL files is in a directory named \INFORMIX, but they actually can exist anywhere in the active file system. DOS environment variables reveal the locations of files. For example, the following statement should be inserted into AUTOEXEC.BAT if the INFORMIX files are in a directory named \INFORMIX:

```
SET INFORMIXDIR=c:\informix
```

An application does not have to be in the same directory as the INFORMIX-SQL files. It, too, may be placed anywhere in the file system. Its location is specified simply by setting DBPATH equal to the directory holding the files.

The program editor built into INFORMIX-SQL is visual and easy to learn, but is not useful for creating much more than simple queries and table maintenance chores. It can handle only eighteen 80-character lines of text. In addition, the editor has several irritating idiosyncracies. It has no commands for

The program editor built into INFORMIX-SQL is visual and easy to learn, but it is not useful for creating much more than simple queries and table maintenance chores.

cursor movement by word left or right, nor any that move the cursor to the beginning or end of the current line. It defaults to the overtype mode and forgets from one invocation to the next that the user has requested insert mode. The backspace key (and ^H) produces a nondestructive cursor movement rather than the more common delete-character-left, which requires an explicit delete key operation.

Use of an external editor is permitted via an environment variable that is called DBEDIT. Within the menu system of INFORMIX-SQL, the Use-editor option is displayed in several different locations. A user who selects this option can run the external editor (the default is EDLIN) instead of the built-in editor.

RDS also thoughtfully provides a DOS interface. Typing **!cmd**, where **cmd** is any DOS command or external program accessible via the PATH variable, runs the command. It is patterned after the UNIX shell escape and, when the command completes, prompts the user for a key-press before returning to the INFORMIX-SQL module.

If the host system contains at least 512KB of memory, then typing **!command** will run another instance of COMMAND.COM, permitting indefinite DOS operation. Returning to INFORMIX is as simple as typing **exit**. This is not documented, however.

The external utility programs that can be called from the ISQL menus also may be invoked directly from DOS without running ISQL. These utilities are used to create forms and reports. Unlike RDSQL commands and queries that are run interactively in interpreted mode, reports and forms are compiled for protection and speed.

SFORMBLD and SPERFORM are the utilities that prepare and execute forms. (The leading *S* differentiates these SQL versions from earlier INFORMIX versions.) To prepare a form, the user first must write the form description file, which contains several sections: database selection, screen layout, a list of tables, a specification of field attributes, and a set of instructions. All but the last one are required. Some assistance can be obtained from the SFORMBLD utility in creating a first draft of the description file, but considerable editing and some programming is required to produce anything but the most trivial forms.

An example of the automatically generated form is shown in listing 1, AUTH.PER. The command to create it from the DOS prompt is:

```
sformbld -d
```

The program will ask for the database and table information. Here, the responses are **edit** and **author**, respectively. It then asks for the name of the form file (the extension .PER is automatically applied). The reply in this case is **auth**. The SFORMBLD module creates a default form description and compiles it, producing a file called AUTH.FRM. In the resulting description file, the column names are used literally, the screen fields are arranged one per line, and no special field attributes are applied. A fair amount of editing was necessary in order to produce the final

TAS™

the "Hard-nosed" Relational Database

$199.

the Database + Integrated Accounting

$349.

If You Liked DBase III, You'll Flip Over TAS™

Hard-nosed business owners have been asking for the power of DBase III™ and RBase 5000,™ but without the high price. That's why Business Tools, Inc. created TAS.™

Compare TAS with DBase III and RBase 5000. You'll see why we think TAS is the best "Hard-nosed" value around. TAS includes a data dictionary. TAS includes a true procedural language. TAS includes a run-time compiler. Plus TAS can be upgraded to multi-user for $100.

TAS lets you develop your own professional menu-driven business applications. And not just simple ones either. TAS applications can hold up to 17 million records. And because TAS compiles them, they run fast.

	TAS™	DBASE III	RBASE 5000
Multi-User	$100	No	$1400
Data Dictionary	Yes	No	Yes
Procedural Language	Yes	Yes	Yes
Compiler	Yes	No	Yes
Records Per File	Unlimited	Unlimited	Unlimited
Files Open	16	10	40
Fields Per File	Unlimited	128	400
File Size	Unlimited	Unlimited	Unlimited
Record Size	10,254	4,000	1,530
Field Size	254	254	1,530
Prices	$199	$696	$700

And TAS is no new comer. TAS was written by Phil Mickelson, the same "Hard-nosed" business person who created The Sensible Solution™ (relational database) in 1982.

TAS comes with a 30 day money back guarantee (less handling fee of $15 Level 1, $30 Level 2, $45 Level 3 and $10 Multi-User). TAS is available for the IBM PC/XT/AT,™ Compaq,™ AT&T 6300,™ Tandy 1000, 1200, 2000™ and most MS-DOS,™ CP/M™ and MP/M™ systems. Multi-user and LAN versions also available.

Get TAS Accounting Software for $349

But why stop with just a database? For just $349, you get TAS Relational Database/Language plus General Ledger, Accounts Receivable and Accounts Payable. And for $499 you get all the above plus Inventory, Sales Order Entry, Purchase Order Entry, and Payroll.

Source code is included FREE! So you can even modify the accounting to fit your business.

Call Today 1-800-648-6258
Ask for operator 44

Call our Toll-Free Hotline. Use your VISA or MasterCard to order today. Outside Washington call 1-800-648-6258. Washington residents call 1-206-644-2015.

Dealer Inquiries Welcome.

NOT COPY-PROTECTED

TAS™

"Affordable Software For Your Business Needs"

Describe your computer system:

☐ 8 bit
CP/M, MP/M
TurboDOS

☐ 16 bit
MS-DOS
PC-DOS

Disk size is:
☐ 5¼″ ☐ 8″

Computer name/model is: ____

Name: ____

Street ____

City: ____

State: ____ Zip: ____

Phone: ____

Quantity:

___ TAS Database/Language–$199

___ Level 2 Accounting (TAS, GL, AR, AP)–$349

___ Level 3 Accounting (Level 2, SO, PO, INV, PAY)–$499

___ Multi-user upgrade–$100

Subtotal (WA add 7.8% tax) ____

Shipping (USA–$5.00 ea.) (Foreign–$20.00 ea.) ____

Total Enclosed: ____

Payment: ☐ VISA ☐ MC ☐ Cashier's Check
(All amounts are payable in U.S. funds only.)

Credit Card Expiration Date: ____

Card Number: ____

Name on card: ____

Signature: ____

MasterCard VISA

4038-B 128th Ave. SE
Suite 266
Bellevue, WA 98006

BUSINESS TOOLS INC.

4038-B 128th Ave. SE
Suite 266
Bellevue, WA 98006

specification file shown in listing 2, M_AUTH.PER.

The default data field marker on a screen is a blank area surrounded by square brackets. Reverse video can be added to this, for example, to distinguish required fields from optional fields. The delimiter can be changed to some other set of characters using the DELIMITERS instruction. INFORMIX-SQL does not support underline and blink attributes or the line-drawing characters of the extended IBM character set.

Intuitive cursor keypad commands allow the user to move around a form easily. A Ctrl-C command cancels the current operation and terminates the use of the form without any modification. Pressing Esc signals the start of a query based on a partially filled-in form. (A blank form used in a query means select all rows.)

When a description file exists (see listing 2 for an example), it is compiled by SFORMBLD to produce an object file with a .FRM extension, which may then be executed by SPERFORM. It may also be called from within the ISQL module using a menu selection and pointing method. The object modules execute quickly. The binary object modules provide developers with a measure of protection against piracy because the source code does not have to be distributed to end users.

The development of a form can be a tedious process, but the results usually justify the effort. Although INFORMIX-SQL does not have a data dictionary for screen fields, the effect can be obtained by careful design of the attributes and instructions in the description file. All data-entry fields can be qualified (range and value checked, cross-referenced to other tables), restricted on input or update, marked required (they are optional by default), verified (the user must retype input to confirm critical data), and marked read-only.

INFORMIX-SQL has no intrinsics for controlling color; further, it usurps the eighth bit of each character so that the line drawing characters are not available.

The conversion of INFORMIX-SQL to the PC is incomplete in the area of screen handling. Forms cannot be designed using color or the IBM line-drawing characters. INFORMIX-SQL has no intrinsics for controlling color and it usurps the eighth bit of each character so that the line drawing characters are not available. Also, RDS has not eliminated the snow that occurs when the color card memory is accessed for update during refresh cycles. The program does work well and quickly on the IBM monochrome display.

The forms-generating abilities of INFORMIX-SQL are flexible, but with that flexibility and power comes complexity. Learning the form conventions and syntax of the description language takes time. Figuring out how to use the composite join feature (linking tables on multiple columns) is difficult; neither the reference manual nor the demonstration programs contains examples.

The SACEPREP utility prepares reports, and SACEGO executes them. These two utilities are used in the same way as the forms programs with one exception—the lack of automatic first-cut report description, which must be accomplished with a program editor and a great deal of manual effort. The report specification file contains several sections: database selection, definitions of parameters and variables, input specifications, output specification, selection criteria, and report format. Numerous commands are provided to control all aspects of report production. A built-in language has looping and conditional branching constructs and functions to perform aggregate calculations.

The ACE utility produces columnar and Ctrl-Break reports in addition to straight line-at-a-time reports. INFORMIX-SQL makes a modest provision for control codes to be sent to the printer via the ASCII statement. The key word *ASCII* followed by a numeric expression emits the character that is represented by the number. This process permits the program to issue control character sequences. A programmer then has the ability to prepackage control strings that can be printed by name as needed to control a printer.

Unfortunately, syntax differences between the screen form and report specifications create confusion. In one INFORMIX-SQL module (the form specification shown in figure 1), some blocks must have explicit termination lines and others must not—the key word of a new block terminates the current one. In the report generator (figure 2), every block requires an end statement. The user is aided by the fact that the forms

FIGURE 1: *Form Specification Structure*

```
database
        [database-name]
screen
{
        .
        .
}
end
tables
        [tbl1 tbl2 ...]
attributes
        [ list of attributes for all display/entry fields ]
instructions
        [ list of instructions ]
end
```

In INFORMIX-SQL's form specification module, some blocks must have explicit termination lines while others must not; the key word of a new block terminates the current one.

FIGURE 2: *Report Specification Structure*

```
database
        [ database-name ]
end
define
        [ parameters, variables ]
end
input
        [ prompt for and accept user input  ]
end
output
        [ set page length, width, and such  ]
end
select
        [ name tables, columns, select and order rows, etc. ]
end
format
        [ specify what the report will look like  ]
end
```

In the report specification module, each block is required to have an explicit end statement. The difference in syntax between this and the form module can be quite confusing.

module, which has the variable syntax, can be forced to create a template automatically, precluding the need to memorize a rule and its exceptions.

DESIGNING PROGRAMS

A user-designed menu option permits customized menus in applications programs. Nineteen levels of menus may be created, each having one or more columns of selections. The selections can be linked to program modules that run when the selection is chosen. Applications may be command driven, menu driven, or a combination of these approaches to the user interface.

The primary debugging help that INFORMIX-SQL offers to programmers is error messages from the command interpreter or the form and report compilers. Some of the messages are useful and to the point, others are not. All displayed messages are numbered and refer the user to documentation; unfortunately, the messages in the appendix are identical to the displayed messages.

The compilers (SACEPREP and SFORMBLD) check for errors and create files with the same base name as the related source files and a .ERR extension if any errors are found. The error file contains a copy of the original listing with annotations pointing to the error location and describing the problem. This is a reasonably good way to help the developer determine what is wrong with a source file. In addition, runtime errors are flagged and described in usually meaningful error messages. If the source file is error-free, an executable file is produced, with a .FRM extension for a form and .ARC for reports.

For the single-user DOS version of INFORMIX-SQL only, access permissions and record-locking features are not implemented. The manual pages describing these features are included in the reference manual, but each is marked with a message that the statement or function is irrelevant in the single-user DOS setting. The affected statements and functions include:

GRANT—specify access privileges to database tables
LOCK TABLE—temporarily block access to a table to other users
REVOKE—remove user-access privileges to a table on an individual basis
UNLOCK TABLE—reverse the effect of a prior LOCK TABLE statement

All of these commands lock whole tables only, not individual records. They are available in the UNIX-based and network versions of INFORMIX-SQL.

A record of transactions on any table in a database may be maintained

using the CREATE AUDIT statement. Typically, at predefined times in the existence of a database, tables are backed up and fresh audit trails are initiated. This makes the recovery of corrupted tables possible. New audit trails are stored in files that can be named by the user or the applications developer. Old audit trail files are cleaned out using the DROP AUDIT statement.

The CHECK TABLE and REPAIR TABLE statements work together to permit restoration of a table that might have been corrupted by a power interruption or other calamity. The checking step determines whether a table or its indexes are damaged. If the indexes are the problem, the REPAIR TABLE statement runs a module called the BCHECK utility, which attempts to restore the B+tree indexes to healthy condition. The RECOVER TABLE statement works in tandem with the audit trail to recover a table from a backup copy made at the time the audit trail file was started. Any inconsistencies cause the recovery operation to stop, meaning recovery is only partially successful.

Indexes are updated automatically each time a table is modified. The RDSQL statements CREATE INDEX and DROP INDEX add and delete indexes at will. The number of indexes on a given table has no practical limit.

Data items may be searched for on indexed and nonindexed columns, but table joins that produce temporary composite tables require indexed columns in the common regions of the joined tables. Composite indexes may be formed on up to eight columns and may be a maximum of 120 bytes wide.

C INTERFACE

RDS offers a companion to INFORMIX-SQL called INFORMIX-ESQL/C (for embedded SQL/C). It is a complete C language interface package that permits SQL statements to be embedded in C programs

ESQL/C is a complete C language interface package. It provides a set of C language functions to be used in INFORMIX forms and reports, enhancing considerably the power and flexibility of the data manager.

and provides a set of C language functions to be used in INFORMIX forms and reports, enhancing considerably the power and flexibility of the data manager. ESQL/C currently works with the Lattice C compiler and will be made compatible with other C compilers if sufficient demand develops.

ESQL/C uses header files, custom preprocessors, and additional subroutine libraries to permit the insertion of SQL statements in normal C programs. Virtually all of the standard RDSQL statements and functions are supported. A knowledge of C programming in general and of structures and pointers to structures in particular are prerequisites for the ESQL/C programmer.

Embedded RDSQL statements fall into four broad categories. The first two, data definition and data manipulation, are self-evident. The other two, cursor management and dynamic management, require some explanation. Cursor, in the context of cursor man-

agement, is really a row pointer. The current row of the active set is the row on which the cursor is positioned in a set of rows produced by a SELECT operation. Dynamic management deals with the fact that RDSQL statements may not be static in a given application, but may be input by the user when the application is running. The tools provided accept user input, determine the type of statement, and execute recognized statements within the running program.

C functions may be called from within report and form specification files and compiled by SACEPREP or SPERFORM. The functions either may be user-defined or standard C library functions. They must be defined in a section of the specification file. A function that returns a value is simply named to run it, while those without return values are invoked by CALL statements. ON BEGINNING and ON ENDING control blocks in the instruction portions of form specification files may be used to specify start-up and clean-up actions that are to be handled entirely by C functions.

The programmer using the C tools within ACE and PERFORM description files must endure the usual compile and link steps typical of high-level language compilers and must prepare a C driver program that will be the execution shell for the report or form. Good instructions and examples are presented in the manual to help programmers get started with ESQL/C.

The package is provided on three disks and is installed in the same way as INFORMIX-SQL (see below). The assumed location of the Lattice C compiler is in \LC, but a DOS environment parameter can change the default.

END-USER CONSIDERATIONS

The INFORMIX-SQL system is provided on four diskettes. They are not copy-protected. When installed on a hard disk, the programs will run without a boot disk in drive A:. A matching serial number and activation key (a sequence of typed characters provided by RDS) are used to validate the installation should the need for vendor support arise.

Installing INFORMIX-SQL is an easy matter; an installation program does practically all the work. The description of the procedure is straightforward and accurate. Some behavioral modifications may be made by setting DOS environment parameters. These are explained in the manual where appropriate. They can be set or changed at any time. Reasonable default values prevail if the parameters are not set.

Through its ISQL module, INFORMIX-SQL provides a powerful user interface for ad hoc queries and database table maintenance. The ISQL module is menu-oriented with the look and feel of an electronic spreadsheet such as Lotus 1-2-3 or Microsoft's Multiplan. The RDSQL interface, a nonprocedural, interactive query language is one of the user's primary means of accessing data. Queries are typed in using the internal editor or an external editor on demand. An inquiry may be run, modified, saved, and recalled easily and quickly.

The SQL SELECT statement is a very powerful query mechanism that permits cross-table joins, multiple selection criteria, subqueries, and a wide range of other options. The general form of a SELECT statement is:

```
SELECT clause
   FROM clause
   [WHERE clause]
   [GROUP BY clause]
   [HAVING clause]
   [ORDER BY clause]
   [INTO TEMP clause]
```

The user specifies the columns from which data values are to be retrieved. The FROM clause names the tables involved in the query, and the WHERE clause is the filter—it specifies the conditions that rows must meet to be added to the SELECT output table. A missing WHERE clause effectively selects all rows of the searched table(s).

The items in the SELECT clause may be more than just column names. Entries may include expressions involving columns, such as summations, max and min functions, etc. The WHERE clause, if present, may be a complex set of expressions and may include subordinate queries, ordering by column values, etc. (For more about SQL and SELECT, see "SQL on the IBM PC," Clyde Holsapple, *PC Tech Journal*, November/December 1983, p. 72).

A SELECT statement may include subqueries, cross-table searches through joins, ascending and descending ordering of responses, arbitrary groupings of information, and summaries. Only the SELECT and FROM clauses are required in a SELECT statement.

Properly designed data forms and reports can be created for custom applications to serve as the only access to data, thus preventing casual users of the application from harming the underlying data.

SELECT requires knowledge of table and column names, available through the Info menu selection of ISQL. This option gives the user access to information about each table, such as its column names and types, index names and types (plain or unique), and status (row length, number of rows and columns). SELECT also requires the user to understand relational concepts, including how tables are joined, how to form meaningful selection statements, and how to use ordering and grouping clauses to format output.

Full-screen data forms and reports may be run either from the ISQL menu or in stand-alone mode. Properly designed data forms and reports can be created for custom applications to serve as the only access to data, thus preventing casual or inexperienced users of the application from accidentally harming the underlying data. Use of intermediate transaction files, which are posted to the main database under controlled conditions, offers further protection against damage to the database.

Data import and export features are available from the RDSQL module. The default delimiter is the vertical bar, but a DOS environment variable, DBDELIMITER may be used to change it, typically to a comma. The feature is rather limited; some problems with data import were experienced. These problems are described below in the section on the sample application.

SAMPLE APPLICATION

The sample application specification provided by *PC Tech Journal* requires the preparation of forms, reports, ad hoc queries, and a set of benchmarks. Table maintenance forms were created to learn the process of form preparation in INFORMIX-SQL. Initially, these forms involved one table each, which simplified the task quite a bit compared to preparing the editorial inventory form, which involved several complex multi-table interactions. The author table maintenance form, called M_AUTH.PER (see listing 2), was later modified to perform a verify-join in order to qualify the state abbreviation entry against a table of state names and abbreviations.

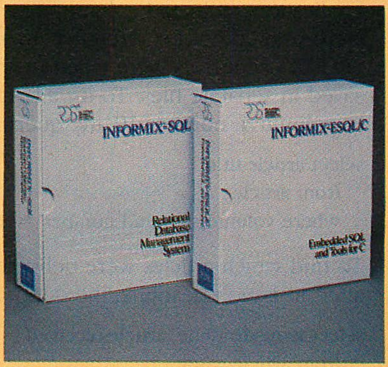
INFORMIX-SQL offers a wide range of form-design features that permit the applications developer to create forms of any description. They may be confined to a single screen or spread across multiple screens that may be paged. Master-detail relationships may be established between tables so that the user may switch from the controlling, or master, table to a subordinate, or detail, table to get a different view of the data. A simple key press switches contexts, and all of this is controlled by a menu bar at the top of the form display.

Data entered into screen fields may be qualified upon input in one of a variety of ways. The verify-join, mentioned earlier for the state abbreviations, uses the LOOKUP attribute. A PICTURE field attribute also can be used to constrain input to any letter, digit, or character on a position-dependent basis. This method was used to gather telephone and Social Security Number data. A FORMAT attribute controls the displaying and inputting of DECIMAL, FLOAT, SMALLFLOAT, and DATE data types.

INFORMIX-SQL OVERVIEW

INFORMIX-SQL, version 1.10.05
Relational Database Systems, Inc., 4100 Bohannon Drive, Menlo Park, CA 94025; 415/322-4100

Product type. Relational database management system with elaborate form and report capabilities.
Software environment. Runs under DOS, version 2.0 and later, UNIX (all versions), and XENIX.
Hardware environment. IBM PC, PC/XT, PC/AT, and compatibles, as well as most hardware capable of running UNIX or operating systems derived from AT&T UNIX.
User interface. The main user interface is menu-oriented. Interrogation of databases is handled by SQL commands that may be created, saved, edited and run from the main menu. Previously compiled forms and reports also may be selected and run from the menu screens.
Help facilities. An on-line help facility displays screens of context-dependent information extracted from the written tutorial user guide.
Printer support. Generic printing to LPT1 is standard, but the report module may include printer control



strings. A DOS environment variable sets the printer port selection.
Form and report design. Detailed specification files are created to describe the appearance and actions of forms and reports. The files are then compiled to produce the executable forms and reports that may be run independently or under the control of the ISQL menu module.
Limitations. INFORMIX-SQL imposes no limits on the number and size of tables, rows per table, columns per row, and number of indexes per table in a database. The host environment (amount of disk space and addressable memory) sets the limits. Composite indexes are limited to eight fields (up to 120 bytes) and rows many contain up to 32KB.
Applications development. INFORMIX-ESQL/C, a separate package, provides the tools to include SQL queries in C programs and to permit the use of C language functions in ACE reports and PERFORM screens. More than 250 third-party applications based on RDS database products are listed in a catalog from RDS.
Data Compatibility. The program reads and writes files in a delimited format not entirely compatible with that used by some DOS applications.
Delivery. UNIX versions since 1981; DOS version since April 1985.
Prices. DOS version: INFORMIX-SQL, $795; network version, $1,995; ESQL/C, $595; C-ISAM, $225; File-it!, $295; XENIX version: INFORMIX-SQL, $995; ESQL/C, $749; C-ISAM, $319; File-it!, $379.
Support. Software support agreement with hot-line telephone assistance and other services is available for a fee. Availability of support is keyed to the customer keeping up with current releases.

—AH

The INCLUDE attribute lets the form designer list a set of valid responses. The user cannot leave the field (except by aborting entry with ^C) until a valid input is typed. The COMMENT attribute allows a display of the set of expected responses on the status line at the bottom of the form.

Other attributes include REVERSE (for reverse video), REQUIRED (an entry must be supplied in update and add operations, but not in query operations), RIGHT (right-justify the display in the field), and UPSHIFT/DOWNSHIFT (convert to upper/lowercase letters upon input). INFORMIX-SQL also has attributes available to control the format and response of screen forms to user input and the presentation of displayed data. Attributes may be stacked in comma-separated lists in the lines that relate screen fields to data in tables.

Reports are handled in a similar way. Listing 3, MLABELS.ACE, shows how a set of mailing labels can be produced for the authors whose articles appeared in a single magazine issue. In this listing, the issue volume and number were hard-coded into the report, but a PROMPT FOR statement could be used to gather that information from the report user at runtime.

Note that the report modules, SACEPREP and SACEGO, use the VARIABLE statement to store temporary values. The variables used in this report hold the buffered name and address strings. The modules allow users to specify arguments on the command line to control program behavior or provide data input. A PARAM statement specifies the variables that will receive the command line input. The total number of variables that can be declared in PARAM and VARIABLE statements is 100 per report specification.

The output specification of an ACE report includes statements to direct output to a file (REPORT TO filename) or a printer (REPORT TO PRINTER). It also has statements to control page presentation (LEFT MARGIN, RIGHT MARGIN, TOP MARGIN, BOTTOM MARGIN, and PAGE LENGTH). The companion FORMAT section controls the appearance of a report within the defined page layout. Its statements include PAGE HEADER, PAGE TRAILER, FIRST PAGE HEADER, ON EVERY ROW, ON LAST ROW, BEFORE GROUP OF, and AFTER GROUP OF. These statements give the designer complete control over the final appearance of a report. The information produced in the report is extracted from the database tables using SELECT statements that work like those of the interactive query module.

Reports use FOR and WHILE loops, IF..THEN..ELSE branching, value assignments, and other procedural language constructs to control their actions and appearance. NEED tells the ACE modules how many lines are needed at the bottom of a page. PAUSE permits page-at-a-time printing. SKIP and SKIP TO TOP OF PAGE also affect formatting.

The ASCII statement used with the PRINT statement emits the ASCII character equivalent of numeric values and is frequently used to send control sequences to printers and screens. The CLIPPED modifier strips trailing blanks from character strings. A set of aggregate expressions and functions perform counting and averaging. The USING expression controls the printed format of dates and numbers.

The ad hoc queries are easy to write because of the power and flexibility of the SELECT statement. No pro-

gramming is required to prepare the queries, although it is necessary to know table and column names. For this application, table and column names were printed out for reference, then the SELECT statements were typed in and saved in separate files. To list all articles in volumn 4, number 10, the query is:

```
select article.title
   from article
   where volume = 4 and number = 10
```

To find which articles were delivered after an issue's deadline, the query is:

```
select issue.deadline, article.received
   from article, issue
   where
   article.volume = issue.volume
   and article.volume = 3
   and article.number = issue.number
   and article.number = 2
   and received > issue.deadline
```

The second query gathers data from two tables. The lines **article.volume = issue.volume** and **article.number = issue.number** cause the two tables to be joined into one composite table for the duration of this query only; therefore, columns in one table can be compared to columns in the other.

No programming is required to prepare any of the queries, although it is necessary to know table and column names.

INFORMIX-SQL exhibits speed in loading and indexing tables (see table 1). Its 95-second time to load sequentially, then index on two columns, is the fastest time measured for any of the data manager products reviewed by *PC Tech Journal* so far (see the August, September, October and November 1985 issues). The RDSQL commands used to load and index are:

```
load from "author.unl"
   insert into author
create index i_author
   on author (last_name, first_name)
```

The elapsed time to index on a two-column key is in the middle of the pack at 42 seconds. In normal operation, reindexing does not have to be a separate task. Add and update operations cause automatic reindexing.

Aggregate functions in SELECT statements simplify queries that require totals. To document and tally codes from the state field in the author table, the RDSQL statements are:

```
select author.state, count(*)
   from author
   group by state
```

Temporary variables and decision logic are unnecessary. The other benchmarks are equally trivial to write in RDSQL.

Some problems were encountered while importing (LOADing in INFORMIX-SQL terminology) a file that had been produced in the ASCII-delimited format and while exporting data (UNLOADing) to a file. Quotes are not treated specially. In the delimited format used by most databases, quotes are used to surround strings that may contain the character that separates the individual columns or fields of a record. INFORMIX-SQL simply passes the quotes through as text characters. If they are stripped out, as they must be, the embedded delimiter character will throw off the column count and affect the reading of the record. If a column contains city and state names spelled out completely and the comma is the delimiter, then Washington, D.C. causes an error.

Another problem resulted from a null character field (nothing between the delimiters) being interpreted incorrectly at the end of a row. INFORMIX-SQL treats the delimiter as a column *terminator*, not as a *separator*. A character field at the end of a delimited row must be a minimum of a single space followed by the new-line character, or else it must have a trailing delimiter to be recognized by INFORMIX-SQL.

A third problem occurred while exporting data, terminating the rows with the delimiter. Other programs that read the file will determine that it has one more column than it actually has and will fail to read the data correctly.

Finally, INFORMIX-SQL expects columns that hold numeric data to have a value explicitly stated, even if it is 0. A null column will not suffice, because that is not valid numeric data. Because some database programs and other applications read and write data in this way, INFORMIX-SQL should be able to read them. A C program was used to convert one of the files provided by *PC Tech Journal* to a form that is acceptable to INFORMIX-SQL. A good progam editor with search and replace also could be used, however tediously, with the same results. The problem is finding the null numeric fields that must be converted to real 0s.

SLIGHTLY TARNISHED

Based on most measurements of performance and desirability, INFORMIX-SQL has a high rating. The package offers more than most users will ever learn how to use in practice. The tools provided to create forms and reports have nearly all of the extra features a serious developer needs. Only the lack of a memo/note data type, support for some kind of arrayed data, and a partly deficient data import/export feature detract from the product's technical appeal.

INFORMIX-SQL shows its maturity in the area of reliability. No bugs were found. The product's behavior is predictable as soon as the user becomes familiar with the concepts and methods used to store, retrieve, and manipulate data. Users experienced only in file management systems will have to learn to think in relational terms to use the product to its potential.

The INFORMIX-SQL package provides some good tools for applications program development. In particular, it has excellent data validation features built into the forms utilities that permit very tight control over what data will be accepted by programs. Using the IF . . THEN . . ELSE action in form instructions permits easy coding of optional fields, such as requesting a department title if the article category is Department. Otherwise the field is blanked out and skipped.

INFORMIX-SQL is fast. Its B+tree foundation provides quick access to data and the SQL user interface has been extended in useful ways.

The shiny luster of INFORMIX-SQL is slightly tarnished by inconsistencies in syntax among modules. In addition to the block-coding syntax differences noted earlier, the product lacks a consistent file-naming convention. In some situations, specifying an extension on a file name is optional, whereas in others it is verboten and in still others it is required. From an end user's or a programmer's perspective, files should always have a default extension that can be overridden.

For those who rank visual appeal high on their list of evaluation criteria, INFORMIX-SQL falls short. A dab of color and easy access to the special IBM graphics characters would help it compete against other database products for the PC. Color and graphics have not been in the forefront of UNIX development because of the way the operating system has evolved to work with even the least-well-endowed terminals.

As far as documentation for INFORMIX-SQL is concerned, the most pressing need for improvement is in providing clearer explanations of complex interactions among tables in a database and how to use some of the more esoteric features. The index is complete but confusing to use, due to its layout, which indents subordinate material from its higher level entry; some subordinate blocks span several pages. The manual combines a user guide and a reference manual with several appendixes in a single volume containing hundreds of pages. Only three dividers serve as signposts to the reader. Another level of tabbed dividers on major chapters within the sections would make the manual easier to use.

INFORMIX-SQL brings a relatively mature and full-featured relational database management system to the microcomputer environment at a reasonable price. The product is well-suited to the development of demanding and complex applications that require data integrity, firm control of the user interface, and high reliability.

Augie Hansen is the owner of Omniware, a software development and training firm that specializes in UNIX and DOS systems and applications. He has written several UNIX-related articles for PC Tech Journal. *The author's net address is: {allegra,amd, attunix,cbosgd,ucbvax}!nbires!bdaemon!arh.*

TABLE 1: *Benchmark Results*

BENCHMARK TASK	TIME (secs)
Add 900 records to an empty database table	95
Index table on two fields (7 bytes)	42
Document and tally codes from one column	29
Mass change one column (28 rows of 900)	14
Extract selected records to create a text file	13

The performance of INFORMIX-SQL is speedy in loading and indexing tables. Its 95-second time sequentially to load then index on two columns is the fastest time measured for any of the data managers *PC Tech Journal* has reviewed so far.

LISTING 1: AUTH.PER

```
database edit
screen
{
last_name       [f000                   ]
first_name      [f001         ]
address         [f002                   ]
city            [f003               ]
state           [a0]
zip_code        [f004 ]
work_phone      [f005         ]
home_phone      [f006         ]
soc_sec_no      [f007        ]
biography       [f008                                        ]
                [f009                                        ]
                [f010                                        ]
                [f011                                        ]
}
end
tables
author
attributes
f000 = author.last_name;
f001 = author.first_name;
f002 = author.address;
f003 = author.city;
a0 = author.state;
f004 = author.zip_code;
f005 = author.work_phone;
f006 = author.home_phone;
f007 = author.soc_sec_no;
f008 = author.biography[1,50];
f009 = author.biography[51,100];
f010 = author.biography[101,150];
f011 = author.biography[151,200];
end
```

LISTING 2: M_AUTH.PER

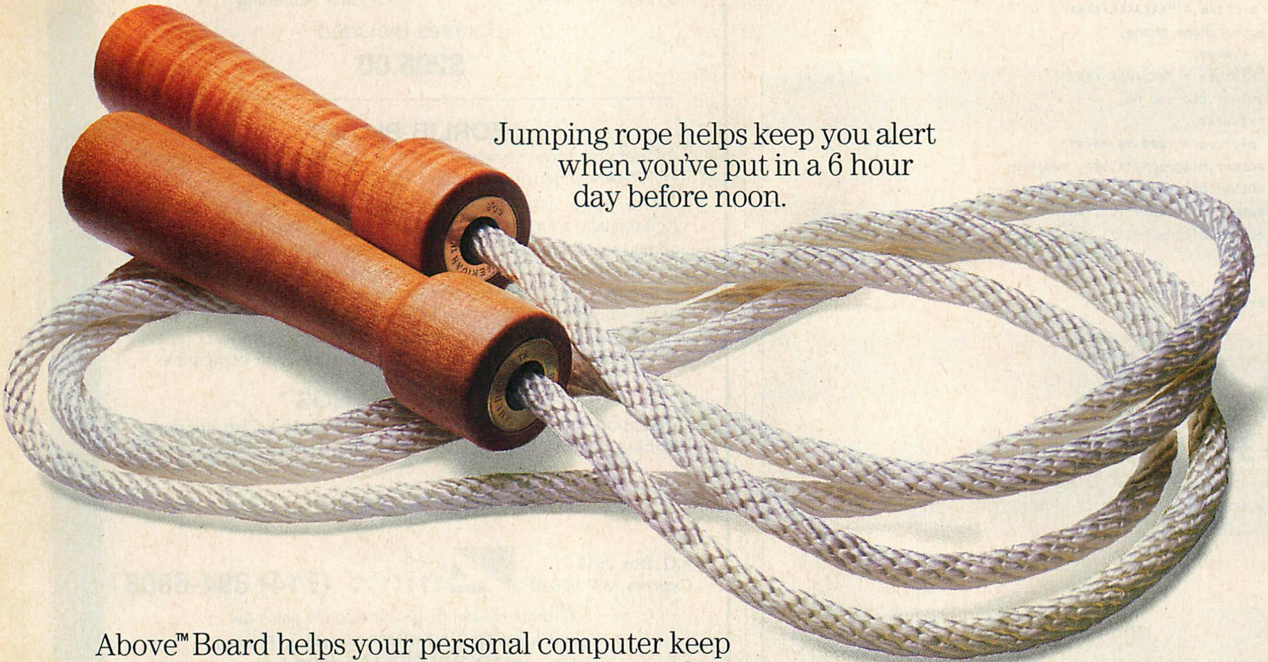
```
database
         edit

screen
{
-----------------------------------------------------------------------

                    AUTHOR FILE MAINTENANCE FORM

-----------------------------------------------------------------------
          Name [f000               ] [f001          ]
                (last)                (first)

       Address [f002               ] [f003          ] [a0]  [f004 ]
                (street)              (city)          (state) (zip)

    Work Phone [f005       ]            Home Phone [f006        ]

Soc. Sec. Number [f007      ]

     Biography [f008                                        ]
               [f009                                        ]
               [f010                                        ]
               [f011                                        ]
-----------------------------------------------------------------------
}
end

tables
        author
        states

attributes
        f000 = author.last_name,
               reverse, required;
        f001 = author.first_name,
               reverse, required;
        f002 = author.address, reverse;
```

```
        f003 = author.city, reverse;
        a0 = author.state,
                reverse, upshift,
                lookup joining *states.abbrev;
        f004 = author.zip_code,
                reverse;
        f005 = author.work_phone,
                reverse,
                picture = "XXX/XXX-XXXX";
        f006 = author.home_phone,
                reverse,
                picture = "XXX/XXX-XXXX";
        f007 = author.soc_sec_no,
                reverse,
                picture = "###-##-####";
        f008 = author.biography[1,50], reverse;
        f009 = author.biography[51,100], reverse;
        f010 = author.biography[101,150], reverse;
        f011 = author.biography[151,200], reverse;

instructions
        delimiters "  ";
end
```

LISTING 3: MLABELS.ACE

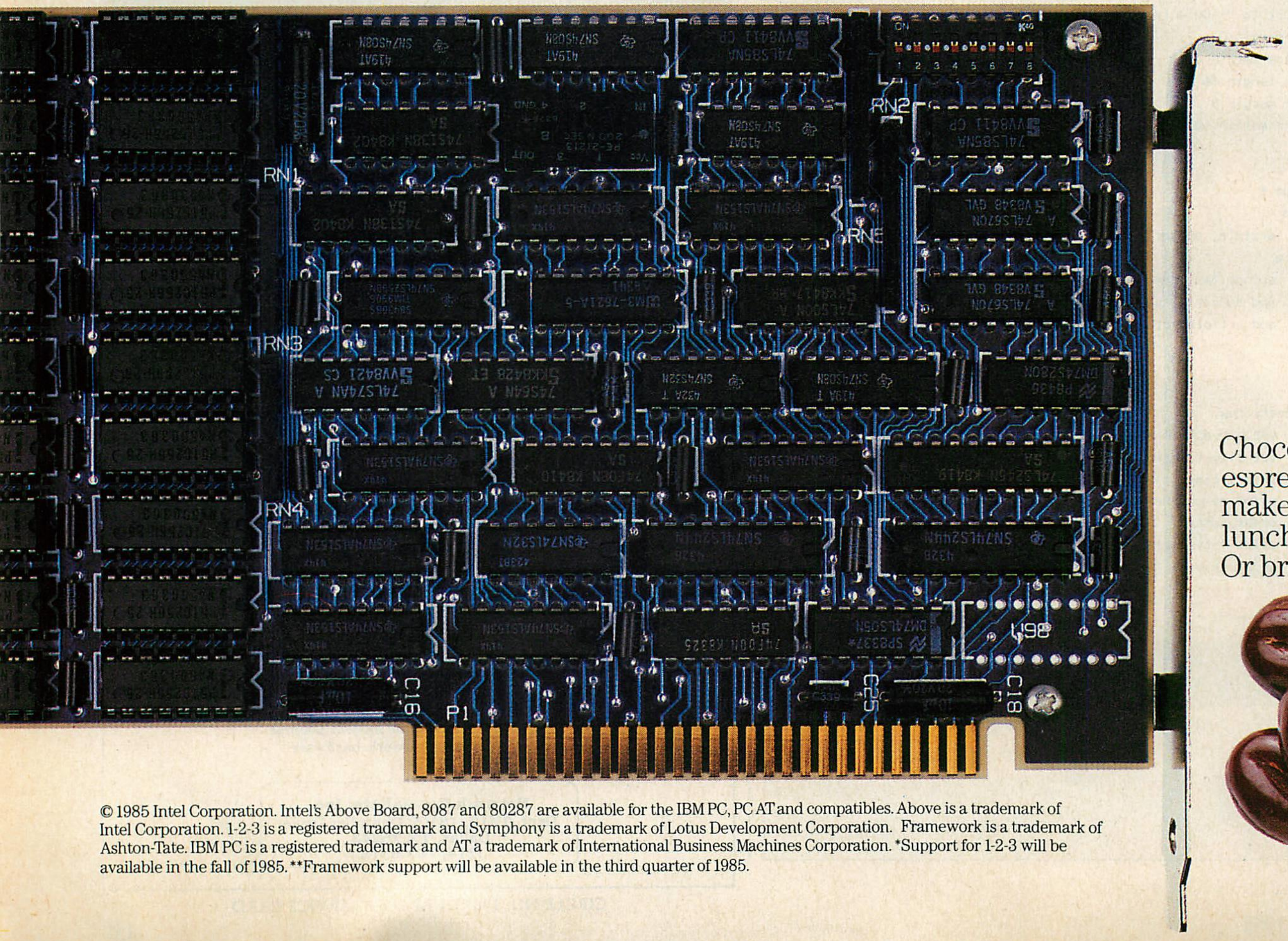
```
{ This report prints mailing label in one column }
{ Assumes labels are 1" high by 3-1/2" wide }

database
        edit
end

define
        variable namebuffer char(40)
        variable citystatezip char(40)
end

output
        report to "labels.txt"
        top margin 0
        bottom margin 0
        page length 66
        left margin 0
        right margin 40
end

select unique *
        from
             article, author
        where
             author.last_name = article.author_lname
             and author.first_name = article.author_fname
             and article.volume = 3 and article.number = 1
end

format
        on every row
             let namebuffer = first_name clipped, 1 space, last_name
             let citystatezip = city clipped, ", ", state, 1 space, zip_code
             skip 1 line
             print namebuffer
             print address
             print citystatezip
             skip 2 lines
end
```

Survival kit for



Jumping rope helps keep you alert when you've put in a 6 hour day before noon.

Above™ Board helps your personal computer keep up with you. It can take you as far as four megabytes beyond 640K when you need it most: Working with Symphony™ release 1.1, 1-2-3® release 2,* or Framework™** (Where only the fittest survive.)



Chocolate covered espresso beans make a nice quick lunch. Or dinner. Or breakfast.

AT Powe

High performance, high capacity hard disk subsystems

A Single DOS Volume up to 240 formatted MBytes in size on your PC is only the beginning of Emerald subsystems' features. However, it is a very important one to many of our customers.

Most serious micro computer users are familiar with the infamous *32 MByte DOS Barrier*. Some of you have only read about it; others have run up against it head on when trying to run PC Focus or a downloaded mainframe program.

If you are one of those who have hit it head on you'll be pleased to know that *for almost a year Emerald has been shipping fixed disk subsystems that solve your problem*. Subsystems. Plural.

All Emerald subsystems can be operated as a single volume up to their formatted capacity. That includes our 36, 50, 70 and 140 MByte subsystems, in addition to the 280.

```
C>chkdsk

237633536 bytes total disk space
    65536 bytes in 2 hidden files
    49152 bytes in 2 user files
237518848 bytes available on disk

   262144 bytes total memory
   205152 bytes free
```

Actual printout of CHKDSK on 240 MByte volume.

You Determine how many volumes exist in your PC, and what size they are. You can have as many as *24 volumes*, and make each one exactly the size it needs to be.

Set-up is menu driven and as simple as *"How many do you want"* and *"How big should this one be?"*

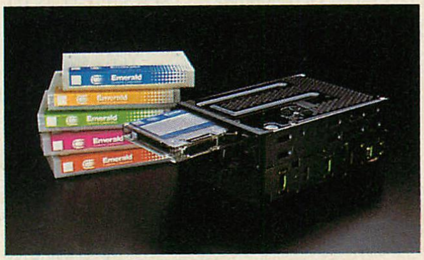
Integrate your existing hard drive into your new sybsystem. Emerald's Disk-Meld technology makes it possible for your XT's 10 Mbyte, or AT's 20 Mbyte drive to become part of a single large volume. For example, if you have an AT with a 20 MByte drive and an Emerald subsystem with a 70, you can use *all* your storage as a single 90 MByte volume. Disk melding makes it easy to combine Emerald drives with each other or with your existing drive to get a subsystem with the storage capacity you need.

Operating System and Network flexibility is yours for the asking. DOS 2.X, and 3.X will get you started. But, your Emerald subsystem can also support multi-user operating systems such as XENIX, VENIX, QNX and PC/IX and networks such as Novell, Sytek, Ungermann-Bass, 3 COM, X-Net, 10 Net, DNA Systems...

Additional Features include password security, 30 millisecond average access time, *automatic retreat to a safe landing zone* in the event of power failure or shutdown, and a long list of PC and AT compatible micro computers.

High performance 1/4" tape backup

60 MBytes in 12 minutes is *FAST* backup, but there's more. If your files are larger than 60 MBytes, Emerald's Backup and Restore Utility (BRU) software will automatically break your file into 60 MByte sections and prompt you for a new cartridge. Of course, restoring is just as easy.



Compact tape drive fits in the AT's front panel expansion space. 60 MByte cartridges are certified for high performance and supplied with color coded labels.

Emerald subsystems were designed for the PC, AT and compatibles such as the AT&T 6300 and Compaq DeskPro.

Menu driven software makes it simple, even for novices, to backup or restore exactly what is needed, and no more. Choose one or more files that were modified after a *Specified Date and Time*, one or more *Specific Files or Directories*, or *All Files and Directories on a DOS Logical Volume*.

Restore data on a different micro if you like. Backup up your company's Emerald subsystem in Portland, Maine and ship the tape to your office in Los Angeles. Because of the defect mapping technique used on Emerald hard disks, the subsystem in L.A. will import your data error free.

Special defect mapping technique allows data to be restored on subsystems other than the original source.

The BRU software *automatically checks, and adjusts to, the defect map* of every Emerald subsystem before restoring data to it.

IBM PC, XT, AT, & PC/IX™ International Business Machines Corp.—AT&T™ American Telephone and Telegraph Corp.—Compaq & DeskPro™ Compaq Computer Corp.—Ungermann-Bass™ Ungermann-Bass, Inc. —Emerald, DiskMeld, BRU & Mainframe Storage for Micros™ Emerald Systems Corp.—UPS™ United Parcel Service.—3 COM™ 3 COM Corp.—DNA Systems™ Network Development Corp.—PC Focus™ Information

r is Yours.

Configured for today's Micros

The Physical Design of the Emerald subsystems lets you determine the configuration that will best serve your needs. Many of today's high performance micros have sacrificed expansion space in favor of compact size; others offer plenty of room for additional drives, tape units and expansion cards. Emerald subsystems provide you with the expansion ability you need. And, they're designed to allow you to continue to expand as your needs continue to grow. No matter which PC you have, there is an Emerald subsystem that will meet your needs.

Subsystems are available for the PC, AT and true compatibles in a variety of configurations.

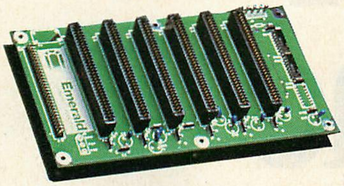
Internal Expansion is easy on the IBM AT and XT. Emerald subsystems are pre-initialized and pre-formatted—just slide the tape or hard drive you have selected into one of the existing expansion areas, plug in a couple of cables, tighten a few screws and replace the system cover. Elapsed time: 10/15 minutes.

The AT accepts 280 MBytes *or* any single drive up to 140 MBytes *and* a 1/4″ tape drive. The XT accepts any Emerald hard drive up to 140 MBytes in size, *or* the 1/4″ tape drive.

External and Portable expansion is easy and practical with the Emerald Portable Subsystems. These IBM color-matched subsystems have their own power supply and are available with hard drives up to 140 MBytes in capacity *or* with a 1/4″ tape drive.

Ordering a tape host adapter card for each of your PCs and physically moving the 1/4″ portable tape sybsystem to where it is needed will save you *thousands of dollars* over the cost of individual portables for each micro.

The portable hard drive configuration is ideal for security sensitive environments. The hard disk is password protected and the entire subsystem is small enough to be locked in a standard safe or filing cabinet.



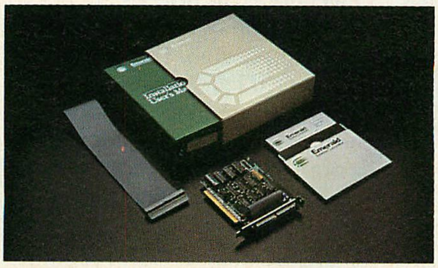
"Expansion chassis" subsystem also provides 6 additional spaces to add cards to your PC.

The 6 Expansion Slots in the Emerald expansion subsystems will be a welcome addition to many micros. If you don't have enough expansion room in your micro, or, if you've used every available slot, then one of these subsystems is just right for you. They are closely matched to the IBM PC in size and color, and have their own, built-in, power supply.

Expansion subsystems are available with drive sizes to 280 MBytes in capacity, with, or without, a built-in 1/4″ tape drive. When coupled with an AT a truly powerful computer system results.

The *real* backup procedure

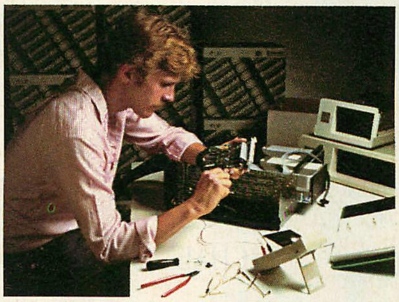
Standing Behind You all the way is the company that broke the 32 MByte DOS barrier.



Documentation, host adapter card, software and cables are included with each subsystem.

First, we provide everything needed to get you up-and-running quickly. Then, we back you up with specialists, available 16 hours a day, to help with network implementation, applications support, XENIX and technical questions, and to provide *48 hour in-and-out repair service.*

Delivery and Warranty are important considerations. Your Emerald subsystem will be shipped within 14 days of order, via UPS, and comes with a 30 day money back guarantee and a 120 day warranty. Warranty extensions are available for 1 and 2 year periods.



Fast service and fast shipping are company policies.

Users with large database applications in factory automation, CAD/CAM, accounting, medical, R&D, Point-of-Sale and many other areas are already telling their friends about Emerald. A large percentage of them are in Fortune 500 companies, and many others are on their way to Fortune status.

If your company fits one of those categories, or, if you're a smaller company with an application you thought could only be done on a mainframe or mini, pick up the phone. Call Emerald. An Applications Engineer is standing by to answer your questions, send you literature and refer you to an installed site in your area.

Let one of our customers tell you, first hand, how Emerald helped give *his* AT Power.

Emerald
Systems Corporation

Mainframe Storage for Micros

4757 Morena Blvd.
San Diego, CA 92117
(619) 270-1994
U.S.A.
Telex: 353458 EMERSYS
Easylink: 62853804

CIRCLE NO. 154 ON READER SERVICE CARD

XENIX™ Microsoft Corp.—VENIX™ Ventur Com—QNX™ Quantum Software—Novell™ Novell, Inc.—X-Net™ XCOMP, Inc.—10 Net™ Fox Research, Inc.
Builders.

A Calendar for the Ages

MICHAEL A. COVINGTON

With these algorithms the user can calculate thousands of years of calendar dates.

Quick, what will the date be 90 days after Christmas 1987? Whether predicting the motions of the planets or figuring out when a loan will fall due, the ability to perform arithmetic with calendar dates is essential to computations involving time.

Calendrical calculations are possible with the help of many makeshift methods of operation. The Pascal routines presented here, however, are noteworthy because they work correctly for calculations involving dates thousands of years to either side of the present date. After installing them in a standard function library, the user can work with them as necessary without worrying about their limitations.

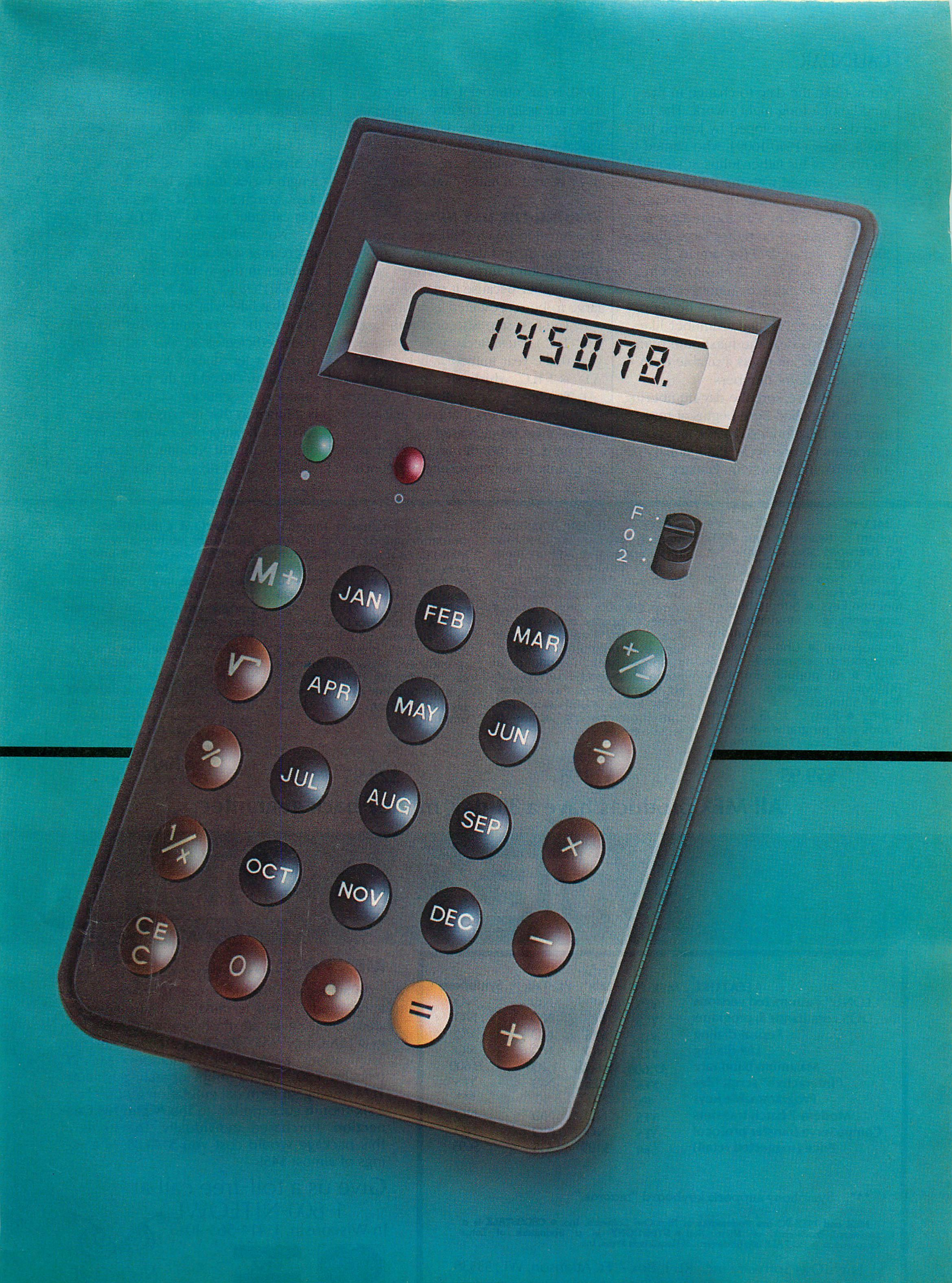
These algorithms are capable of producing valid results for calculations involving any date from 4713 B.C. through at least 5000 A.D. when run on an IBM PC using Turbo Pascal. The degree of accuracy depends on the number of significant digits represented in the computer's floating-point (real) data type. The algorithms that are applied here were taken from Jean Meeus' *Astronomical Formulae for Calculators*, but they have been published in many other places as well. Astronomers have been using them for years.

DAY NUMBERS

The secret to handling dates conveniently is to adopt a system of day numbers. Instead of working with a date expressed as a year, month, and day, a single number can be used to indicate how many days have elapsed since a particular reference point, or *epoch*. After the desired calculations are finished, the day number can be converted back to year, month, and day.

Computer programmers have a tendency to refer to any day numbering system as a *Julian date*. Strictly speaking, the Julian date is the number of days that have elapsed since noon on January 1, 4713 B.C., a date chosen to facilitate easy conversion between various ancient calendar systems. This day begins at noon because the Julian date is used primarily by astronomers who

ILLUSTRATION • ANDY LEVINE


145078.
F
O
2
M+
JAN
FEB
MAR
APR
MAY
JUN
JUL
AUG
SEP
OCT
NOV
DEC
CE
C

do not like the date to change in the middle of a long night's work. The Julian date was developed by Joseph Justus Scaliger (1540-1609), who named it in honor of his father Julius; it should not be confused with the Julian calendar that was established by Julius Caesar.

The problem with Julian dates is that they tend to be rather large numbers. The Julian date for January 1, 1986, for example, is 2,446,431.5. On most microcomputers, numbers larger than 32,767 or 32,768 cannot be represented as integers, and a number like 2,446,431.5 is near the maximum number of significant digits available even for floating-point numbers.

For this reason, the day numbering system described here begins at midnight on December 31, 1979, an epoch astronomers refer to as 1980 January 0, or 1980.0. In this system, January 1, 1980, is day number 1; days before 1980 are assigned negative numbers. The routines treat the day number as a Pascal real number, but all twentieth-century day numbers are small enough to be stored as integer variables.

FINDING THE DAY NUMBER

The function DAYNUMBER converts a year, month, and day to a day number. The function returns the answer as a real number because it may be too large to be represented as an integer.

The first step is to deal with negative (B.C.) dates. Because there was no year 0, the year immediately before 1 A.D. is 1 B.C., which is represented as −1. To correct for the absence of a year 0, all year numbers that have negative values, must be increased by 1.

Next, the month numbering must be changed so that January and February are treated as the 13th and 14th months, respectively, of the previous year. This places February at the end of the year, which is an advantage because February is the only month whose length varies and that is not 30 or 31 days. March is treated as month 3, and this calendar has no month 1 or 2.

Now, advantage can be taken of an interesting coincidence. If m is a month number in this system, then $INT(30.6001*(m+1))-63$ is the number of days that have elapsed in a given year prior to the beginning of that month. The coefficient is 30.6001 rather than 30.6 in order to allow for rounding errors introduced by the computer's binary representation of numbers. INT is a function that returns the integer part of a floating-point number; INT differs from TRUNC only in that its result is also a floating-point number.

New Version

Sizzling C.

The fastest C. The C that Microsoft developed to write its own software programs. Hot.

So hot that we can make this claim: Virtually every program runs faster with Microsoft® C Compiler than with any other MS-DOS® C compiler.

Efficient C.

We give you everything you need to write code so tight your computer will scream.

"Preliminary testing on the Microsoft C Compiler produced code that was significantly smaller than that produced by other C compilers."
Paul Springer, Ashton-Tate.

"We found the FAR pointer very helpful for situations where a mix of memory models offers the greatest efficiency."
Robert Frankston, Software Arts.

"The portability of the code between MS-DOS and XENIX® is great."
Jim Bean, Peachtree Software.

But it's really no surprise that our C stretches your micro to its limits. We wrote both the MS-DOS and the XENIX operating systems.

*Purchase both Microsoft C Compiler and Microsoft Macro Assembler and get a $25 rebate direct from Microsoft. See package for details.

For the name of your nearest Microsoft dealer, or to upgrade from Microsoft C Compiler or Lattice C, call (800) 426-9400. In Washington State, Alaska, Hawaii and Canada, call (206) 828-8088.

And make your programs really cook.

MICROSOFT.
The High Performance Software™

Microsoft C Compiler Version 3.0

Microsoft C Compiler
- Produces compact code and fast executables.
- Implements register variables.
- Small, Medium and Large Memory model Libraries-Mix models with NEAR and FAR pointers.
- Transport source and object code between MS-DOS & XENIX operating systems.
- Library routines implement most of UNIX System V C library.
- Choose from three Math libraries and generate in-line 8087/287 instructions or floating point calls.
  - —Floating point emulator (utilizes the 8087/287 if installed).
  - —8087/287 coprocessor support.
  - —Alternate math package—extra speed on systems without an 8087/287.
- Link routines written in Microsoft FORTRAN (V 3.3 or higher), Microsoft Pascal (V 3.3 or higher) or Microsoft Macro Assembler.
- Supports MS-DOS pathnames and Input/Output redirection.
- File sharing and record and file locking is supported.
- Do source level debugging, with the Symbolic Debug Utility, available separately with the Microsoft Macro Assembler Package*

Library Manager
Create, organize and maintain your object module libraries created with Microsoft languages.

Object Code Linker
- Simple overlay linker combines relocatable object modules created using Microsoft Languages into a single program.
- Link very large programs (over 1MB, using overlays).

EXEPACK Utility
A new utility to compress sequences of identical characters from an executable file and to optimize the relocation table.

EXEMOD Utility
A new utility used to modify the fields in the header according to the instructions given by the user in the command line.

'C' Benchmarks—done on a Compaq Plus with 512k memory with no 8087. Program "SIEVE," with register variables,

	Exec Time	Code Size	EXE Size
Microsoft C	:9.39	141	5,914
Lattice C	:12.24	164	20,072

Finally, the day number can be calculated. It is a function of *y*, the adjusted year number; *m*, the adjusted month number; and *d*, the day of the month. The day number for the beginning of the current year can be found with *FLOOR* (365.25**y*), where FLOOR is a function that returns the nearest whole number not greater than its argument. For example, *FLOOR* (−3.5) is −4 (note that *INT* (−3.5) is −3).

The term *INT* (30.6001*(*m*+1)) calculates the first day of the current month. The current day of that month, *d*, is added next. Finally, the term −723,244.0 is added. This does two things: first, it subtracts 63 as required by the second term; second, it subtracts 723,181 so the day number will be reckoned from the beginning of 1980, not the beginning of 1 B.C. (or 0 A.D.).

BACK TO GREGORIAN

Thus far this routine has performed calculations only with the Julian calendar. If, however, the calculated day number is equal to or greater than −145,068, the Gregorian calendar should be used instead. The difference between the Julian and Gregorian calendars affects calendrical computations. In the original system designed by Julius Caesar, any year divisible by 4 was a leap year; thus, the average length of a year was 365.25 days. In 1582, Pope Gregory changed the equations slightly. In his system, a year that is divisible by 100 is not a leap year unless it is also divisible by 400, and the average length of the year becomes 365.2425 days. This is much closer to the earth's actual orbital period, which is 365.2422 days.

Next, the month numbering must be changed so that January and February are treated as the 13th and 14th months, respectively, of the previous year.

The switch from Julian to Gregorian calendars is easily accomplished. First, the number of leap year days that should have been left out must be determined. Then necessary corrections must be made for them.

The changeover date, or the date the Gregorian calendar was adopted at

Non-stop flights to XENIX.

The XENIX® market is taking off. To keep up you've got to convert your MS-DOS® software to run on the new, faster 80286 machines. Quickly. Efficiently.

Nobody makes porting software to the XENIX environment as easy as Microsoft; you simply recompile the original source code.

Our full-featured XENIX languages use commands and syntax identical to their MS-DOS siblings. So you don't have to rewrite your program to move it.

Whether your applications are written in Microsoft versions of Pascal, BASIC, COBOL or FORTRAN, we have the solution to your transportation problems.

MICROSOFT®
The High Performance Software™

For the name of your nearest Microsoft dealer call (800) 426-9400. In Washington State, Alaska, Hawaii and Canada, call (206) 828-8088.

Overall features of XENIX Languages
- Shell support of syncronous and asyncronous tasks.
- Support of pipes for interprocess communication as a generalized I/O device.
- Record and file locking:
  - For sequential, random, dynamic and ISAM files.
  - Blocked or unblocked file reads.
  - Deny access, deny read and deny write lock levels.
  - Automatic or manual record and file locking of ISAM files.

Microsoft FORTRAN Compiler
- Based on ANSI 77 (x3.9-1978) standard.
- Single-and double-precision for both real and complex data types.
- Math packages: 287 support, alternative fast math and BCD decimal math.
- Large program support.

Microsoft COBOL Compiler
- High Level 2 ANSI 74 standard features.
- Multi-key ISAM with split keys, alternate keys and duplicate keys.
- Interactive extended screen section.

Microsoft BASIC Interpreter
- Standard Microsoft BASIC.
- Supports trace, single step and immediate execution.
- Supports random, dynamic, ISAM and sequential file I/O.

Microsoft Pascal Compiler
- ANSI and ISO standard with extensions.
- Math packages: 287 support, alternative fast math and BCD decimal math.
- Large program support.

the papal court, has the day number −145,078; if the Julian to Gregorian correction is not applied, however, its day number is −145,068. This corresponds to October 5, 1582, in the Julian calendar but October 15 in the Gregorian. The 10-day jump was intended to correct the error that had been accumulating since Roman times. Individual countries stayed on the Julian calendar much longer; Great Britain and America did not go Gregorian until 1752 (with an 11-day corrective jump), and Russia, Greece, and Turkey kept the Julian calendar until the twentieth century.

When dealing with historical dates, remember also that the Roman custom of beginning the year on January 1 was not always respected. Through much of the Middle Ages, the year began on Christmas Day. In Britain, between the years 1300 and 1752, the year began on March 25; April Fools' Day was then a new year's celebration.

The procedure CALDATE translates a day number back into a year, month, and day. This algorithm is different from DAYNUMBER in that it works with Julian dates throughout. The first step is to translate the day number into a Julian date, *z*. Because, for example, 36,524.25 is the length of a Gregorian century, and 1,867,216.25 is the Gregorian equivalent of the Julian date February 29, 400 A.D. (the date on which the Julian and Gregorian calendars first diverged), corrections must be made for Gregorian-Julian discrepancies.

The heart of CALDATE is a series of numerical tricks similar to the trick with the number 30.6001 discussed above. The final step corrects for the absence of a year 0; if the calculated year is less than 1, then 1 is subtracted from it.

WEEKDAY gives the day of the week, deduced from the day number modulo 7. Many of the day numbers in these routines are too large to be converted directly to integers, and Pascal's MOD operator cannot be used on floating-point numbers. WEEKDAY adjusts for this by adding or subtracting 28,000 (a large multiple of 7) as many times as necessary to bring the day number to within the appropriate range. Then it converts the day number to an integer and invokes the MOD operator.

REFERENCE

Meeus, Jean. *Astronomical Formulae for Calculators.* 2nd edition. Richmond, Virginia: Willmann-Bell, 1982.

Michael Covington conducts research in artificial intelligence and supercomputer applications at the University of Georgia.

LISTING 1: CALENDAR.PAS

```
{ Long-range calendrical package in standard Pascal     }
{ Copyright 1985 Michael A. Covington                   }

function frac(x:real):real;
  { Fractional part of a real number.                   }
  { Turbo Pascal provides this as a built-in function. }
begin
  while x < -maxint do x:=x+maxint;
  while x > maxint do x:=x-maxint;
  frac := x - trunc(x)
end;

function int(x:real):real;
  { Integer part of a real number.                      }
  { Uses real data type to accommodate large numbers.  }
  { Turbo Pascal provides this as a built-in function. }
begin
  int := x - frac(x)
end;

function floor(x:real):real;
  { Largest whole number not greater than x.            }
  { Uses real data type to accommodate large numbers.  }
begin
  if (x < 0) and (frac(x) <> 0) then
    floor := int(x) - 1.0
  else
    floor := int(x)
end;

function daynumber(year,month,day:integer):real;
  { Number of days elapsed since 1980 January 0 (1979 December 31). }
  { Note that the year should be given as (e.g.) 1985, not just 85. }
  { Switches from Julian to Gregorian calendar on Oct. 15, 1582.    }
var
  y,m:   integer;
  a,b,d: real;
begin
  if year < 0 then y := year + 1
            else y := year;
  m := month;
  if month < 3 then
    begin
      m := m + 12;
      y := y - 1
    end;
  d := floor(365.25*y) + int(30.6001*(m+1)) + day - 723244.0;
  if d < -145068.0 then
    { Julian calendar }
      daynumber := d
  else
    { convert to Gregorian calendar }
    begin
      a := floor(y/100.0);
      b := 2 - a + floor(a/4.0);
      daynumber := d + b
    end
end;

procedure caldate(date:real; var year,month,day:integer);
  { Inverse of DAYNUMBER; given date, finds year, month, and day.  }
  { Uses real arithmetic because numbers are too big for integers. }
var
  a,aa,b,c,d,e,z: real;
  y: integer;
begin
  z := int(date + 2444239.0);
  if date < -145078.0 then
    { Julian calendar }
    a := z
  else
    { Gregorian calendar }
    begin
      aa := floor((z-1867216.25)/36524.25);
      a := z + 1 + aa - floor(aa/4.0)
    end;
  b := a + 1524.0;
  c := int((b-122.1)/365.25);
  d := int(365.25*c);
  e := int((b-d)/30.6001);
  day := trunc(b - d - int(30.6001*e));
  if e > 13.5 then month := trunc(e - 13.0)
             else month := trunc(e - 1.0);
  if month > 2 then y := trunc(c - 4716.0)
              else y := trunc(c - 4715.0);
  if y < 1 then year := y - 1
          else year := y
end;

function weekday(date:real):integer;
  { Given day number as used in the above routines,   }
  { finds day of week (1 = Sunday, 2 = Monday, etc.). }
var
  dd: real;
begin
  dd := date;
  while dd > 28000.0 do dd:=dd-28000.0;
  while dd < 0 do dd:=dd+28000.0;
  weekday := ((trunc(dd) + 1) mod 7) + 1
end;
```

Everything You Always Wanted To Know About

APPLICATION DEVELOPMENT DATABASES...

DATA

Just What Is DataFlex, Anyway?

The easiest thing to say is that DataFlex is an application development system *like* d-BASE. However, that wouldn't be a fair statement about either product because DataFlex is not only faster, more powerful and easier to use than d-BASE, but also supports TRUE multi-user transaction processing with complete data integrity. Applications developed with DataFlex can also be run UNCHANGED on a wide selection of 8 and 16-bit operating systems and LANs.

More Powerful and Easier To Use?

Absolutely. DataFlex uses "image formatting" to quickly and efficiently develop input screens and report formats for your application. All you have to do is make an "image" of your screen or form using any ASCII text editor. AUTODEF, DataFlex's file definition utility, automatically generates an error-free, ready to compile data entry program, creates the necessary data and key index files, and makes a data dictionary entry describing each field, its length, type and format. With DataFlex, there's no need for you to go back and provide tedious definitions of the length, format and data type of your windows. And while we're on the subject of text editors, you should know that DataFlex is also available with a full function word processing option that's operationally equivalent to and compatible with MicroPro's WordStar + Mailmerge.

What About A Procedural Language?

DataFlex has a powerful procedural language that combines the best features of Pascal, BASIC and RPG. It has over 125 commands in the following categories:

Argument Processing	Indicators
Console I/0	Key Procedures
Control	Multi-User Functions
Database Commands	Reporting
	Sequential I/O
Definition	String Operations
Data Entry	Structural Control
Forms Processing	System Commands

DataFlex allows programs of up to 2,750 command lines incorporating up to 255 screen images! This enables you to design systems far more complex and sophisticated than possible with some other database programs. For many applications, however, knowledge of the procedural language is not necessary since DataFlex's powerful AUTODEF and QUERY functions automatically generate the source code for handling data entry and reporting. Data checking and formatting commands and error traps can easily be added to the source code before compiling so your applications will run smoothly with minimal possibility of operator error.

How Can DataFlex Be So Fast?

DataFlex utilizes a multi-keyed B+ ISAM structure which updates indexes on-line each time data is entered, deleted or edited. Since all data is instantly available for recall, time consuming key sorts and batch index reorganizations are not necessary. With DataFlex, you'll never again have to wonder whether or not your computer is really working or "hung-up" somewhere in the middle of a sort. DataFlex reports appear on your screen or printer as quickly as the data can be read from your disk. Data input is also speeded by DataFlex's FlexKeys™: single keystroke commands that perform record finding, saving and editing functions. There's even a HELP FlexKey that can summon instructions or explanations pertinent to your application.

Will DataFlex Exchange Data With Other Programs So I Don't Have To Start All Over From Scratch?

Yes! Import and export of data is a snap. DataFlex can read or output either comma or carriage return/linefeed delimited files. Conversion of d-BASE data files is described in detail in our comprehensive user's manual.

The Application Development Database For Single-User, Multi-User And Local Area Networks

FLEX™

What's DataFlex Have For Those Of Us With Heavy Reporting Requirements?

Output for reports, labels, preprinted forms and files is handled by a powerful DataFlex macro command called REPORT. It consists of an integrated set of selectable predefined output routines that can be chosen as needed. These routines are used to "fill-in" an output "image" and then send it to the device of your choice. You can write the report command file yourself or it can be automatically generated by DataFlex's QUERY function. Even complex multi-file reports can be generated through QUERY. All you do, using the arrow keys or a mouse, is "point-and-shoot" at the data you want to see! QUERY then automatically writes error-free source code and allows you to save it as an ASCII file that you can then customize, compile and run. Output can be sent directly to your printer or CRT, or saved as a comma or carriage return/linefeed delimited ASCII file for later use by DataFlex or some other program. The speed with which QUERY performs its source code generation function is something that you have to see for yourself to fully appreciate.

SUPPORTED OPERATING SYSTEMS AND NETWORKS

IBM PCDOS 1.x, 2.x, 3.x
MSDOS 1.x, 2.x, 3.x
IBM "AT" XENIX
IBM PC NETWORK
CP/M, CP/M-86
Concurrent CP/M-86
Concurrent DOS
MP/M-86
Novell NetWare
TurboDos
Corvus w/IBM PC
Molecular N/Star
TeleVideo Infoshare
3Com EtherShare
PC-Net
NorthStar Dimension
Action DPC/OS
DMS Hi-Net
Alloy Engineering RTNX

SPECIFICATIONS

Environment:
- 8080, Z80, 8088, 8086, 80186, 80286

Requirements:
- 52K TPA (8-bit)
- 256K TPA (16-bit)
- CRT w/cursor addressing
- 600K disk storage

Capabilities:
- 255 Database files
- No limit on number of open files (16-bit)
- 9 6-segment indexes per file (16-bit)
- 16K Bytes per record
- 255 Fields per record
- 16,777,215 Records/file
- 2,750 Program lines per configuration
- 255 Screens per configuration
- 2,900 Windows per configuration
- 32,000 Variables
- 127 Indicators
- 9 Global break point levels
- 18 Terminal independent function keys

Does DataFlex Work On MULTI-USER Systems and LANs?

Yes! DataFlex, unlike most other products, supports TRUE multi-user processing. With DataFlex, no user is ever locked out of a file or denied access to a record. Every user can, at any time, access, read and even change any record in the database while maintaining absolute data integrity!

How Easy Is It For The "End User" To Deal With DataFlex Applications?

DataFlex includes an elegant menu system which totally insulates the end-user from the computer's operating system. Each menu screen supports up to nine prompted actions each, including chaining to "sub-menus" and DataFlex programs, and the execution of system commands and other programs. A pre-programmed "help-screen" is included to provide operator assistance on selecting items from the menu. Password security can be established for each menu action to prohibit unauthorized file access, and the passwords (or even the entire menu) can easily be changed at any time by programmers with access to DataFlex's MENUDEF utility.

With Some Programs, I've Had Performance Go Right Down The Tube When There's More Than A Thousand Or So Records. How's DataFlex Perform With Large Databases?

Large, complex databases are exactly what DataFlex was designed for, and its performance in this environment is impressive. In benchmark tests on a Wang PC with a 36,000 record database of 128 byte records, DataFlex was able to find a record via a 41 byte key and display it to the screen in .8 seconds! This high level of performance extends to multi-file operations as well, where the PC version of DataFlex puts no limit on the number of open database files. As many as 255 database files can be maintained by DataFlex, the size of each limited only by your operating system and DataFlex's 16.7 million record per file "limit."

DataAccess Corp., 8525 S.W. 129th Terrace, Miami, FL 33156 (305) 238-0012 TELEX 469021 DATA ACCESS CI

CIRCLE NO. 118 ON READER SERVICE CARD

Come See Us at COMDEX Booth No. 3150

If you need easy access to your IBM mainframe, our Emulation Adapters are just the tickets.

With the IBM Enhanced 5250 Emulation Adapter or the IBM 3278/79 Emulation Adapter, you can quickly convert your IBM Personal Computer into an online terminal. Thereby gaining easy access to mainframe data without leaving your desk. And making the transfer of data between PC and mainframe simple.

What's more, you can access programs at the host and run PC programs simultaneously. Switching from one to the other with a simple keystroke sequence.

You can even pick your own printer in most cases: the host's high-speed printer or your own PC's printer.

Put the power of our most popular mainframes on your desk.

The enhanced 5250 emulation card allows you to access the IBM System/34, 36 or 38 with your PC, PC XT, PC AT or *Portable* PC. (Remote access is available on the System /36 and 38 using appropriate hardware and software.)

And the 3278/79 card lets you and your IBM PC, PC XT or PC AT interact with the IBM 4300, 3081, 3083 or 3090, either local or remote.

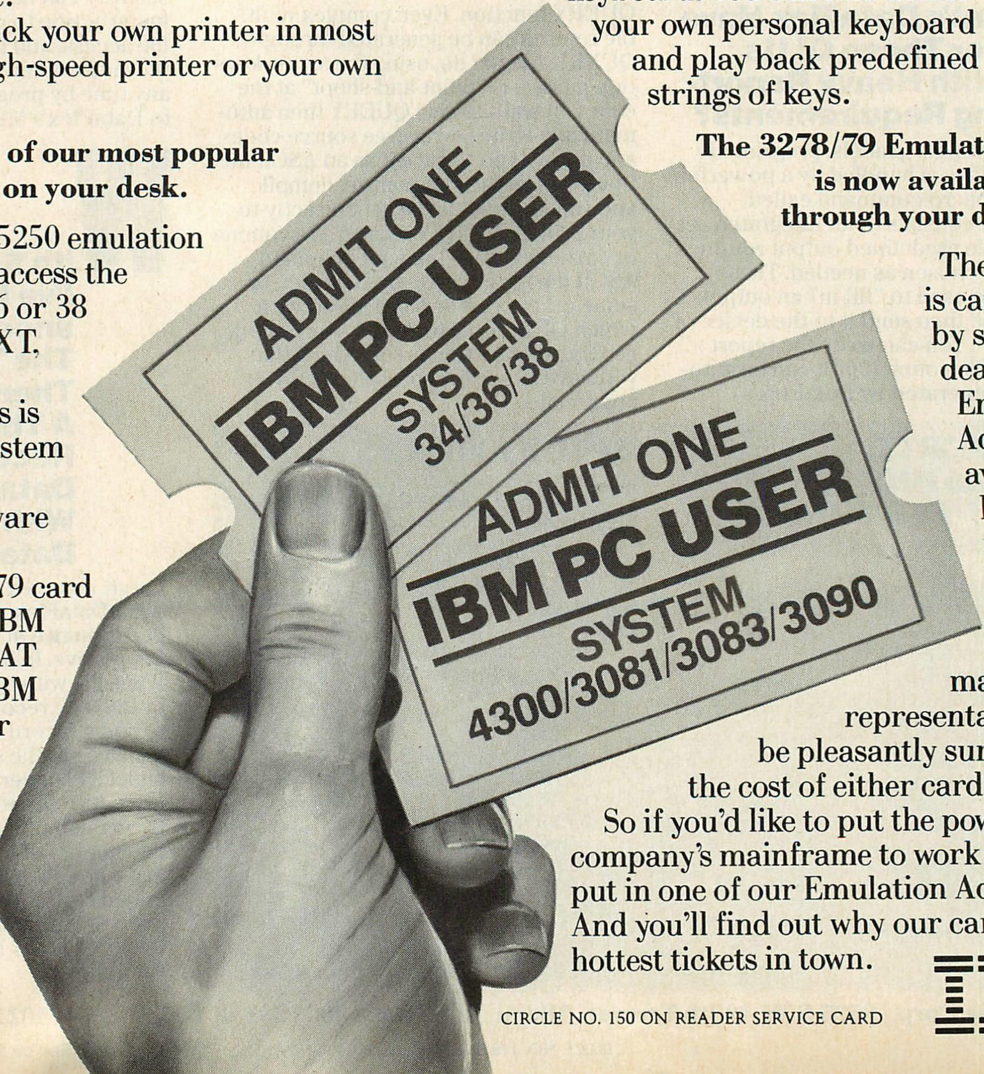
You'll thank us for the memory.

The 3278/79 card's emulation control program requires only 23K of memory.

Which means, with our 3278/79 card, you've got more memory left for application programs. And less tied up with housekeeping functions.

The key to keyboard familiarity.

The enhanced 5250 card's emulation program lets you define keyboard functions. Default definitions make it easy to map the PC, PC XT, PC AT or *Portable* PC keyboards to the 5291 keyboard. You can also define your own personal keyboard layout and play back predefined strings of keys.



The 3278/79 Emulation Adapter is now available through your dealer.

The 5250 card is carried by selected dealers. Both Emulation Adapters are available at IBM Product Centers and through your IBM marketing representative. You'll be pleasantly surprised by the cost of either card.

So if you'd like to put the power of your company's mainframe to work for you, put in one of our Emulation Adapters. And you'll find out why our cards are the hottest tickets in town.

CIRCLE NO. 150 ON READER SERVICE CARD

IBM®

Safe Landing for Hard Disks

ILLUSTRATION • FRANK RILEY

A short program positions hard-disk drive heads over cylinders guaranteed to contain no data and where they can do no harm.

MURRAY LESSER

The common belief persists today that all hard-disk files are fragile devices capable of destroying data at the slightest vibration or jolt. This is not really the case. The IBM-supplied hard disks for the PC/XT system unit as well as the PC expansion unit use Winchester technology. This well-seasoned, rugged technology is distinguished by a low-mass, taper-flat slider that carries the read/write head, a lightly loaded slider suspension system, and a lubricated iron-oxide magnetic coating on the disk surfaces. The combination of the light loading and the lubricated oxide coating on the disks allows the slider to take off from the recording surface, and later to land, as the drive accelerates when starting and slows down at power-off, without causing any damage to either head or data.

The initial commercial application of Winchester technology (the IBM 3340, first shipped in 1973) featured user-replaceable data modules, each of which contained a head-disk assembly. No head-retraction system raised the heads off the surface before the module was removed or replaced.

Under most circumstances, neither disk nor data is in danger if the heads of the XT hard disk accidentally land on

a data track when the system's power is shut off. Nonetheless, IBM supplies a "Prepare System for Relocation" routine on the diagnostic diskette furnished with the XT, the main purpose of which is to position the hard-disk heads over a cylinder guaranteed to be empty. This avoids any possibility of data loss in case of major shocks to the system while it is being relocated.

As an extra precaution, the relocation routine can be run at the end of every session before the power is shut off. This would protect against the possibility of damage from earthquakes or any other disasters that might strike before the drive is restarted. Continually rebooting the diagnostic diskette, however, is a time-consuming and annoying task. A much faster and easier solution involves a short program called LANDER.COM that can reside permanently in the root directory of the hard disk. This program is written in assembly language and runs under DOS. Like the relocation routine, it moves the drive heads to the off-data landing zone where they cannot do any damage.

SOME CAVEATS

LANDER is intended for use *only* with IBM fixed-disk drives installed in the XT system unit or PC expansion unit. It is not to be used with the DASD drive in a PC/AT. Of necessity, LANDER uses the ROM BIOS routines that reside on the XT fixed-disk adapter card.

The program may not work with all IBM-compatible systems or substitute drives. It almost certainly will not work with any drive that does not respond correctly to the IBM-supplied FDISK and FORMAT utilities or that requires an additional installed device driver to operate under DOS.

The DEBUG tests discussed below should be used on IBM compatibles, as well as any non-IBM hard-disk drives mounted in a PC, before LANDER is run. A system that passes these tests should support LANDER. The program itself should be run carefully with DEBUG before live data are used.

When the IBM programmers wrote the ROM BIOS routines for the hard-disk controller, they included four different drive parameter tables in order to allow hardware designers a little room to maneuver. These tables are stored in ROM BIOS at a location given by the address vector for interrupt 41H. According to the listing in the *IBM Technical Reference Manual* (revised April 1983), the setting of the switches on the adapter card dictates which of the four tables should be used. Some systems, however, do not have switches on their adapter cards. In these cases a jumper takes care of the switch settings and automatically fits the installed drive.

The figures referred to below are copies of screen displays produced by the programs noted. The D: drive is a memory-mapped virtual disk that uses the installable device driver listed in the *IBM DOS Technical Reference Manual* for DOS 2.1. D: drive can be used as the default drive to avoid accidental deletion of necessary files and to speed up the compilation and assembly procedures. The user can reach programs stored on the hard disk (which is the C: drive) with a PATH facility established by an AUTOEXEC.BAT routine after the system is booted up.

The debugging procedure shown in figure 1 is necessary for all systems that do not use an IBM drive; it confirms that the ROM BIOS controller has called the appropriate table to the location specified by the interrupt 41H vector. For the IBM adapter, the tables are stored at segment:offset C800:03E7, as shown. (The first row of display has been moved in order to line up equiva-

lent values in all of the four tables, and extraneous ASCII symbols have been removed for clarity.)

Each table occupies 16 bytes. The first word of each table is the number of cylinders in the drive; this is followed by a byte that gives the number of heads. IBM drives use the fourth table and start at offset 0417; thus, they have 132H (306D) cylinders, four heads, and, as a result, four tracks per cylinder.

Cylinders are accessed by number, starting with 0. The last cylinder on the drive should be number 305. LANDER uses subterfuge to place the drive heads over cylinder number 306, a cylinder guaranteed to contain no data.

When a hard disk leaves the factory, it already has been formatted with cylinder, track, and sector numbers; the DOS FORMAT utility, however, does not do this. FORMAT cleans up the File Allocation Table and directory sectors, checks the rest of the DOS partition for unrecoverable read errors, but writes only the system and label information to the drive directory and data areas.

If a hard disk is formatted with the routine provided on the Advanced Diagnostics diskette furnished with the *Hardware Maintenance and Service Manual*, a fixed pattern is written to all available locations on the drive. This will clean up any errors FORMAT marked as nonrecoverable. However, after formatting with this routine, the user must start all over and run FDISK and the DOS FORMAT utility.

The next DEBUG test moves the access mechanism to the highest cylinder described by the parameter table. The procedure is shown in figure 2. This test should not be run until irreplaceable files on the hard disk have been backed up in case of error when the ROM BIOS routines are used.

The program can be written directly to memory, starting at offset 100H, with the DEBUG **assemble** command. If the system has a hard disk, interrupt 13H is the call to the ROM BIOS; for any disk code (as given in register DL) less than 80H, interrupt 13H will eventually find the diskette drive adapter. If, on the other hand, the system does not have a hard disk, interrupt 13H will call the diskette drive adapter directly.

If the user calls on the adapter when AH contains 08, the controller is required to return to the user a description of the installed drive characteristics. The DEBUG command G107 sets a breakpoint after the return from the interrupt and displays the data in the registers. If no errors exist, register

FIGURE 1: *Checking Hard-Disk Parameter Table*

```
D>[DEBUG]
-[DO:104 L4]
0000:0104  E7 03 00 C8
-[DC800:3E7 L40]
C800:03E7                          32-01 02 32 01 00 00 0B 00
C800:03F0  0C B4 28 00 00 00 00 77-01 08 77 01 00 00 0B 05
C800:0400  0C B4 28 00 00 00 00 32-01 06 80 00 00 01 0B 05
C800:0410  0C B4 28 00 00 00 00 32-01 04 32 01 00 00 0B 05
C800:0420  0C B4 28 00 00 00 00
-[Q]

D>
```

The four hard-disk drive parameter tables are checked with DEBUG. The fourth table describes IBM drives; they have 132H (306D) cylinders and four heads.

FIGURE 2: *Moving Hard-Disk Drive Heads*

```
D>[DEBUG]
-[A100]
43AA:0100 [MOV AH,08]
43AA:0102 [MOV DX,80]
43AA:0105 [INT 13]
43AA:0107 [MOV DX,80]
43AA:010A [INC CH]
43AA:010C [MOV AX,0401]
43AA:010F [INT 13]
43AA:0111 [INT 3]                    [Automatic return to DEBUG]
43AA:0112                            [<CR> only]
-[G107]                                   [Display drive parameters]

AX=0000  BX=0000  CX=3051  DX=0301  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=43AA  ES=43AA  SS=43AA  CS=43AA  IP=0107   NV UP EI PL ZR NA PE NC
43AA:0107 BA8000        MOV     DX,0080
-[G]                                 [Move one cylinder past "end"]

AX=0000  BX=0000  CX=3151  DX=0080  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=43AA  ES=43AA  SS=43AA  CS=43AA  IP=0111   NV UP EI NG NZ AC PE NC
43AA:0111 CC            INT     3
-[G=10A]                                  [Attempt next-higher cylinder]

AX=0202  BX=0000  CX=3251  DX=0080  SP=FFEE  BP=0000  SI=0000  DI=0000
DS=43AA  ES=43AA  SS=43AA  CS=43AA  IP=0111   NV UP EI PL NZ NA PO CY
43AA:0111 CC            INT     3
-[Q]

D>
```

The disk-drive head can be stowed with a little work in DEBUG. The first INT 13 instruction reports the hard-disk parameters, while the second INT 13 stows the head.

FIGURE 3: *LANDER.COM from LANDER.ASM*

```
D>[MASM LANDER;]
IBM Personal Computer MACRO Assembler Version 2.00

49950 Bytes free

Warning Severe
Errors  Errors
0       0

D>[LINK LANDER;]

IBM Personal Computer Linker Version 2.20

Warning: No STACK segment

There was 1 error detected.

D>[EXE2BIN LANDER LANDER.COM]

D>
```

The IBM Macro Assembler is used to compile LANDER.ASM. LINK converts MASM's .OBJ file to an .EXE file. EXE2BIN is responsible for creating the final .COM file.

FIGURE 4: *Running LANDER with a Hard Disk*

```
D>[LANDER]

Fixed disk in landing position

TURN OFF MASTER POWER SWITCH
D>
```

The user should check the drive parameters before running LANDER (see figure 2). Turn the power off after running LANDER; a DOS command might undo the program's work.

AH will contain 00; if, on the other hand, an error does exist, AH will show an error code that can be translated from the data in the ROM BIOS listing.

The results of a successful AH = 08 request can be interpreted as follows. The value in DL is the number of hard drives installed (in this case, one). The value in DH is the number of the last head on the drive (the four heads are numbered from 0). The number of 512-byte sectors per track is shown in the low-order six bits of register CL. If the two high-bit locations in CL (bits with the values 8 and 4 in the high nibble) are masked off, the 11H (17D) sectors per track are revealed. The highest available cylinder number is shown in ten bits: the two bits left over from CL followed by the eight bits of CH. The highest available cylinder in figure 2 is 130H (304D). Cylinder 305 is not available to the DOS device driver; it is reserved for diagnostic write tests.

This portion of the test may result in different numbers on a non-IBM drive. If the differences between the drives are only in the numbers of cylinders and heads, and these correspond to the actual values for the system's drive, LANDER still may be applicable. If the differences are more extensive, the user may damage files by running LANDER using the system's drive.

If the first part of this test is successful, the heads can be sent to the write-test cylinder with the next highest number. By placing 0401H into AX before calling on the drive adapter, the user can request verification that the seek address in the registers was found. If no error message is returned, the requested disk address was found (head 0, sector 11 on the cylinder numbered one higher than the highest available). Now, the rest of the code in figure 2 can be executed with the G command (the INT 3 in offset 0111 returns control to DEBUG). After this pass is completed, the final code segment should be run a second time, starting at offset 010A, with the command G=10A. This time, error message 02 should be returned, signifying that the address was out of the range indicated by the applicable parameter table, so the controller did not even try to find it.

The source code for LANDER.ASM is in listing 1. The program is converted to a .COM file using the EXE2BIN utility after it has been linked (see figure 3). The program assembles correctly with Macro Assembler 1.0 or 2.0.

All irreplaceable data should be backed up on another medium before LANDER is run. As LANDER.COM is

Why The Industry Leaders Developed The Enhanced Expanded Memory Specification

On June 17, 1985 at PC Expo in New York, three industry leaders, AST Research, Inc., Ashton-Tate and Quadram Corporation jointly announced the Enhanced Expanded Memory Specification (Enhanced EMS). A result of the combined efforts and engineering know-how of these three companies, it was developed to provide the industry with a more flexible and advanced open architecture for addressing memory beyond the 640K limit imposed by the IBM PC memory map. Other leaders in the industry, including software developers Borland International, Digital Research Inc. and Sorcim/IUS, are in agreement.

Designed as a true superset of the Lotus/Intel EMS, the Enhanced Specification defines both hardware and software interface improvements, which work to allow maximum flexibility in addressing the needs of a broad spectrum of software environments.

AST RESEARCH INC.

ASHTON·TATE

QUADRAM

Q: What improvements does the Enhanced EMS bring to the software development community?

A: The Enhanced EMS offers three main improvements:

1. It supplies additional page registers for accessing expanded memory.
2. It permits the use of a larger total page window area above 640Kb.
3. It allows software the option of direct paging into any available 16Kb block within the 1 Megabyte PC address range.

Q: What are the key advantages of the Enhanced Expanded Memory Specification?

A: The key advantages are:

Enhanced EMS	Lotus/Intel EMS
Supports Lotus/Intel 64K Page Window; additional 16K page windows can be located on any available 16Kb boundary	64Kb Page Window located between 784Kb and 960Kb (8 locations)
Total Page Window area limited only by available memory	64Kb maximum Page Window area only
Multiple windows supported	Supports only one window
64 Page Registers	4 Page Registers
Direct swapping into memory addressed in 0-640Kb Conventional Memory range. (Preferred to be under operating system control.)	Not supported

1. As a true superset, you can optionally utilize the improved functionality of the Enhanced EMS while maintaining compatibility with the Lotus/Intel specification.
2. A special function call is included that enables the application software program to access a total page window area above 640Kb as large as 256Kb, or even larger if memory below 640Kb is used.
3. Only page register contents must be modified to substitute one program environment for another, therefore the time-consuming memory-to-memory moves or memory-to-peripheral operations normally required can be avoided.

This is ideally suited to providing the rapid page swapping associated with some multi-tasking operating system environments.

Q: How were the architectural advantages of the Enhanced EMS achieved?

A: The Enhanced EMS memory mapping algorithm is based upon 64 registers (versus 4 in the Lotus/Intel specification) each representing a 16Kb page in the PC memory space. This general architecture places a minimum of restrictions on the software development community, and allows mapping throughout the entire 1Mb of PC-addressable memory anywhere into the 2Mb of paged memory (64 Registers x 16Kb/Register = 1Mb).

Q: What about mapping into conventional memory (DOS 640K memory limit)?

A: The option of mapping into conventional memory depending upon specific needs or type of application is provided. However, there is no requirement to utilize this feature.

With the two largest PC enhancement product manufacturers providing both hardware solutions and technical support you can be assured your software development investment is protected when you select the extra benefits of the Enhanced EMS. If you have additional questions call our "Enhanced EMS Phone Lines":

AST Research, Inc.
714/863-1333 Ext. 5906

Quadram Corp.
404/923-8612

If your haven't already received your copy of this enhanced open architecture specification, ask for our Specification Kit. A complete package will be mailed to you.

QUADRAM

passed through DEBUG, a breakpoint should be taken after every INT 13 instruction to check the register contents. During this process, the user can test the MOVIT loop in MOVER by repeatedly setting the breakpoint at the JZ MOVIT instruction with a G261 followed by a T instruction. This will return control to the start of the loop.

LANDER sets up a dummy set of drive parameter tables identical to the tables in ROM BIOS with the exception of the number of cylinders; LANDER increases this number by 16 for each table. This subterfuge along with IBM's method of formatting XT drives, which provides track and sector addresses one cylinder beyond the last advertised, allows LANDER to position the drive heads over data-free cylinders where they can do no harm.

The user can check for the first hard disk by attempting to recalibrate the drive. If no hard disk is installed, the attempt will fail (error 01), and the program will end with the "No fixed disks installed" message.

If the recalibration is successful, the MOVER subroutine will be called. MOVER repeats the second DEBUG test described above, this time with better programming practice to avoid potential errors. MOVER advances the cylinder number until an error 40H is obtained, indicating that the access mechanism was unable to proceed (even though the new parameter table claims the cylinder exists). During this failing operation, the computer emits a burping sound, and the drive light stays on for an extended period of time as the controller recovers by resetting and recalibrating the drive.

Some hard disks will advance to the end of the revised table without running off the end. These will produce an 02H error message.

The cylinder address producing the error code should be at least one number higher than the cylinder number that produced the 02 error code on the second DEBUG test (for the IBM hard disk CX should show at least 3351H); otherwise, LANDER is not successfully reaching a safe landing zone.

After the unsuccessful attempt to drive the heads off the end, MOVER decreases the cylinder address and repositions the heads to the landing zone. It restores the original interrupt 41H vector and returns to the main program.

When program control is returned from MOVER, LANDER tests for a second hard-disk drive. If the system does have a second hard disk installed, the MOVER process is repeated. If it does not, error code 80H (meaning attachment did not respond) will be returned when an attempt is made to recalibrate drive 81H. LANDER will display the single-drive message and instruct the user to shut off power, as shown in figure 4.

If LANDER does not operate exactly as described here while running under DEBUG, either an error was made in transcribing the source code, or an incompatible drive and adapter were being used. In either case, the improper use of the ROM BIOS routines may have damaged data on the hard disk.

Despite the widespread belief that all hard disks are fragile devices, Winchester technology, with its lubricated oxide surfaces, is safe to use with personal computers, where poor operating environments and minimum maintenance are the rule. Nevertheless, running LANDER prior to every planned shutdown is a good practice.

Murray Lesser is the author of Using the Microsoft Business BASIC Compiler on the IBM PC *to be published by McGraw-Hill this month. He retired after 26 years at IBM.*

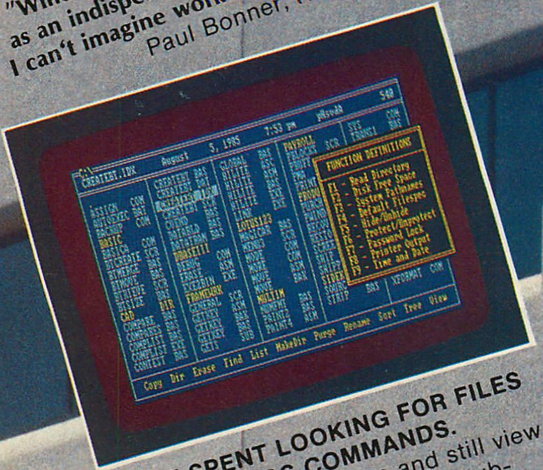
LISTING 1: LANDER.ASM

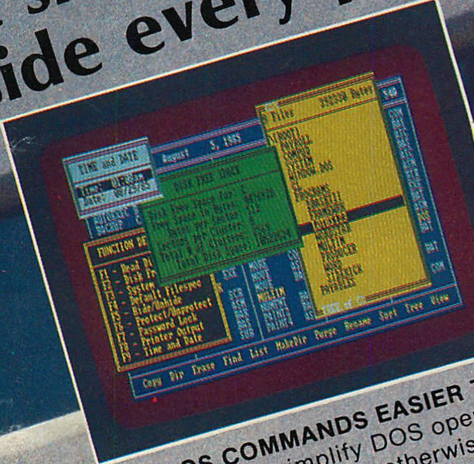
```
        PAGE    ,100
;**************************************************************
;                    LANDER.ASM                               *
;   LANDER.COM is a program to move the fixed disk heads on the*
; IBM PC/XT (or PC) to the "shipping" cylinder (cylinder 306 on*
; the standard fixed disk).  When LANDER displays its closing*
; message, turn off the master power switch to land the heads*
; in a safe position in case the system unit is accidentally*
; moved.                                                      *
;   WARNING:  LANDER may not work properly with "third-party"*
; drives or with "IBM-compatible" systems.                    *
;**************************************************************

;       Written by M. L. Lesser, June 28, 1985
;       Assembled with IBM PC Macro Assembler v2.00
;       Linked file converted to .COM file with EXE2BIN

CODE    SEGMENT PARA PUBLIC 'CODE'
        ASSUME CS:CODE, DS:CODE
        ORG     100H
LANDER  PROC    NEAR
        JMP     DOIT

;Messages:
MESS_1  DB      13,10,'No fixed disks installed',13,10,'$'
MESS_2  DB      13,10,'Fixed disk in landing position'
        DB      13,10,'$'
MESS_3  DB      13,10,'Both fixed disks in landing position'
        DB      13,10,'$'
MESS_4  DB      13,10,'TURN OFF MASTER POWER SWITCH$'

;Local storage:
OLDINT  DD      ?                       ;Original INT 41 vector
PARMS   DB      64 DUP(?)               ;Substitute parameter table

DOIT:
;Set up temporary drive parameter table:
        MOV     AX,3541H                ;Get and store orginal vector
        INT     21H
        MOV     WORD PTR OLDINT,BX
        MOV     WORD PTR OLDINT+2,ES
        LDS     SI,OLDINT               ;Source starting address
        MOV     AX,CS                   ;Destination starting address
        MOV     ES,AX
        MOV     DI,OFFSET PARMS
        MOV     CX,64                   ;Number of bytes to move
        CLD                             ;Move increasing direction
        REP     MOVSB                   ;Move them
        MOV     AX,CS                   ;Restore DS addressability
        MOV     DS,AX
;Increase maximum number of cylinders in new table by 16:
        MOV     AX,16
        LEA     BX,PARMS
        ADD     [BX],AX
        ADD     10H[BX],AX
        ADD     20H[BX],AX
        ADD     30H[BX],AX
;Check for existence of hard drive:
        MOV     AH,11H                  ;Attempt to recalibrate
        MOV     DX,80H                  ; first hard disk
        PUSH    DX
        INT     13H
        OR      AH,AH                   ;Was command performed OK?
        LEA     DX,MESS_1               ;Set up no-drive message and
        JNZ     ENDIT                   ; exit if no fixed drives
        POP     DX
        CALL    MOVER
;Check for existence of second hard drive:
        MOV     AH,11H                  ;Attempt to recalibrate
        MOV     DX,81H                  ; second hard disk
        PUSH    DX
        INT     13H
        OR      AH,AH                   ;Did it do it OK?
```

WindowDOS™ Interactive DOS Utility Program

Puts DOS where it should be-- Inside every program!

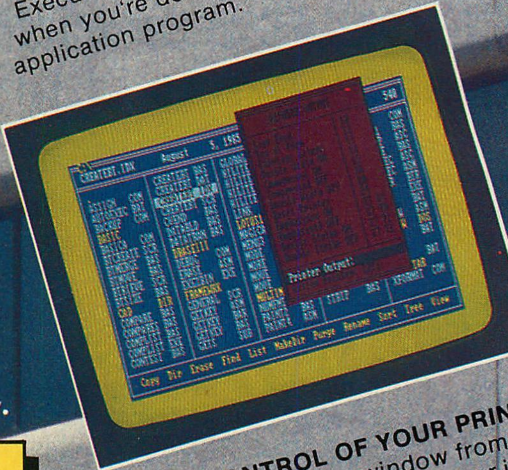
"WindowDOS quickly established itself as an indispensible part of my PC environment. I can't imagine working without it."
Paul Bonner, Reviewer, PC Week





MAKE DOS COMMANDS EASIER TO USE.
Pop-up windows simplify DOS operations and add functions and features not otherwise available. Execute commands with a single keystroke. And when you're done, return instantly to your application program.

REDUCE TIME SPENT LOOKING FOR FILES AND PERFORMING DOS COMMANDS.
See up to 85 files per screen page and still view file size, creation time and date. Identify sub-directories at a glance--display contents by moving cursor to directory name and pressing ENTER.



TAKE FULL CONTROL OF YOUR PRINTER.
Call up the Printer Output window from inside any program and select the type font or issue any other printer command. With more than one parallel printer on your system, select the one you want.

Satisfaction guaranteed or we'll return your money.

Compare WindowDOS with its competition in these 20 important categories!

	WindowDOS	SideKick	1 Dir	XTREE	Starbridge DOS	Norton Utilities
Works inside another program	Yes	Yes	No	No	No	No
Display Directory	Yes	Yes	Yes	Yes	Yes	Yes
Sort Directory	Yes	No	Yes	Yes	Yes	Yes
Find any file on hard disk	Yes	No	No	No	No	Yes
Copy, erase or rename file	Yes	No	Yes	Yes	Yes	No
Selective file erase command	Yes	No	No	No	No	Yes
Batch file copy or erase	Yes	No	Yes	Yes	Yes	No
Change file attributes	Yes	No	No	Yes	No	Yes
Change default subdirectory	Yes	No	Yes	No	Yes	Yes
Create new subdirectories	Yes	No	Yes	Yes	Yes	No
Display Visual TREE	Yes	No	No	Yes	No	No
Check free space on drive	Yes	No	Yes	Yes	Yes	Yes
Display file in ASCII format	Yes	No	Yes	Yes	Yes	Yes
Display file in HEX format	Yes	No	No	Yes	No	Yes
Check or set time and date	Yes	No	Yes	No	Yes	Yes
Send printer control codes	Yes	No	No	No	No	No
Select default printer	Yes	No	No	No	No	No
Password "lock" computer	Yes	No	No	No	No	No
Screen timeout feature	Yes	No	Yes	No	No	No
Windows	Yes	Yes	No	No	No	No
Copy protected	No	No	No	Yes	Yes	No
Cost	$49.95	$84.95	$95	$49.95	$49.95	$99.95

Memory resident (The feature is available from inside another program.)

What SideKick doesn't do, WindowDOS does. They work together as perfect companions. No other utility can compare with the many fine features found in WindowDOS!

WindowDOS

$49.95
Plus $5 Handling/Shipping
1-800-433-5355
Texas 214 264-2626
Money orders, cashier's checks, VISA, MC, AmEx, and corporate purchase orders.

Software of the Future, Inc.
P.O. Box 531650 Grand Prairie, Texas 75053

CIRCLE NO. 127 ON READER SERVICE CARD

WindowDOS Interactive DOS Utility Program for IBM PC, XT, AT and true compatibles. Requires DOS 2.0 or newer and 80 column display. Uses only 40K of memory. All brand names listed above are registered trademarks.

```
        LEA     DX,MESS_2       ;One-drive message
        JNZ     ENDIT           ;Exit if no second drive
        POP     DX
        CALL    MOVER
        LEA     DX,MESS_3       ;Two-drive message
ENDIT:  MOV     AH,09H          ;Display "drive" message
        INT     21H
        LEA     DX,MESS_4       ;Display final message
        MOV     AH,09H
        INT     21H
        INT     20H             ;and quit

MOVER:                          ;Move to highest track on disk
;Get original drive parameters:
        MOV     AH,08
        PUSH    DX
        INT     13H
;Install new vector for INT 41:
        MOV     AX,2541H
        LEA     DX,PARMS
        INT     21H
;Initialize drive parameters with new table:
        MOV     AH,09
        POP     DX
        PUSH    DX
        INT     13H
;Move head past highest pre-formatted cylinder:
MOVIT:  CLC                     ;Clear carry flag
        ADD     CH,1            ;Next cylinder
        JNC     M2              ;If register overflowed
        ADD     CL,40H          ; increase high bits
M2:     MOV     AX,0401H        ;Move and verify sector
        INT     13H
        OR      AH,AH           ;Past end?
        JZ      MOVIT           ;If not, try one more
;Return to shipping cylinder:
        CLC
        SUB     CH,1
        JNC     M3              ;In case of underflow
        SUB     CL,40H          ; decrease high bits
M3:     MOV     AX,0401H
        INT     13H
;Restore original INT 41 vector:
        LDS     DX,OLDINT
        MOV     AX,2541H
        INT     21H
;Restore DS addressability:
        MOV     AX,CS
        MOV     DS,AX
;Initialize to original drive parameters:
        MOV     AH,09
        POP     DX
        INT     13H
        RET

LANDER  ENDP
CODE    ENDS
        END     LANDER
```

Low power

Complete hard disk kit $395

Comes complete with virtually the identical controller that's in the IBM® XT, and Xerox® warranties the hard disk for one year

Guaranteed 100 percent IBM PC compatible

How can we offer this fantastic price? Simple. We buy in such volume that even the most avaricious hard disk businessmen understand they have to give us the best price possible. We could pocket the difference, but we don't.

Instead, we put the extra profit into our testing facilities. That's why Xerox guarantees our $395 10 megabyte hard disk for one year.

Xerox knows, as our customers know, that we have an extensive testing program. Here is what we contribute toward giving you the maximum hard disk performance.

Best Drives Available

First, we buy the best drives available. Sounds trite, doesn't it? I mean, a drive's a drive—right? Hardly. You should see some of the junk we get in our labs. Some have such high failure rates that we even questioned our own $10,000 hard disk tester. But when we tested other manufacturers' drives we were assured that our equipment was fine, which just confirmed that the bad hard disks were not only bad—they were real bad.

But that's just the weeding out process. We then take each drive that we've put through our tester and test it again with the controller you've requested. We call this a "tested pair."

DOS Doesn't Do It

In case you're thinking that all this is an unnecessary duplication of what DOS does for you, let me explain the disk facts of life.

If DOS did what you may think it is supposed to do when you format the disk, DOS would map around these bad areas. Unfortunately, DOS doesn't do this.

DOS 2.0 and 2.1 can't enter the bad tracks. DOS 3.0 can, but only on the IBM AT. Unfortunately, as the press has so well documentated, the AT's hard disk develops bad tracks later on.

We do what DOS can't

We believe the problem is so bad, we use a software program that performs a powerful test of your disk drive on all of the IBM or IBM compatible computers—PCs, XTs, and ATs. Our format takes hours to analyze the disk. But when we finish, you know that the bad tracks are really mapped out so you won't write good data that will disappear into a black hole. We even send you a printed statement of our test results.

Our software allows you to type in the bad track locations from the list supplied by the manufacturers, so you'll never write good data to them—even if DOS didn't identify them as bad. The software even lets you save the location of these bad sections to a file, so that you can reformat your disk without spending hours retesting.

We even include a program that will give you continuous comments on the status of your hard disk. No more waiting for that catastrophic failure.

Average Access Time

As you might suspect, some hard disks are faster than others in their ability to move from one track of data to another. The time it takes the hard disk to move one-half way between the beginning of the disk to the end is called the "average access time."

The first generation of 10 megabyte hard disks had average access times of 80-85 milliseconds (msec). But computer users love speed, and guess what—the average access time for the new 20 megabyte hard disk in the IBM AT is only 40 msec. (We sell an AT equivalent with only 30 msec access time!)

There are some legitimate reasons for the shorter access time. It's particularly helpful when there are multiple users on the same hard disk. It's also important when running a compiler. But remember, before you get too wrapped up in the access speed, there's always that ST 506 interface which won't let data transfer from the hard disk to the computer any faster than 5 megabits/second. We've bypassed that choke hole, too. If you want the functional equivalent of a Ferrari with a turbocharger, order our 10 Mbit per second 100 megabyte hard disk with 18 msec of average access speed.

Compatibility

To be sure that your hard disk is 100 percent compatible with the IBM XT you don't need to buy the same hard disk that's in the XT. You can't even be sure *what* brand hard disk it is because IBM, like Express Systems, goes into the marketplace and buys hard disks from several vendors. However, they buy their XT hard disk controller from only one vendor—the same one we do.

You can buy the IBM XT controller from IBM for $495 or you can buy from us, the functional equivalent, manufactured by the same company that makes it for IBM for only $195. Is it the exactly identical IBM XT controller? No, it's better. First, it takes less power, and secondly, it can control from 5 to 32 megabytes—the IBM controller can work with only 10 megabytes. It is 100 percent IBM XT compatible, and 100 percent is 100 percent. If you want to save a slot, we carry a version that lets you operate two hard disks and two floppy disk drives.

More than 32 Megabytes

You can operate with more than 32 megabytes (the limit of DOS) through the use of "device drivers." Express Systems can supply you with device drivers for our hard disks for over 32 megabytes formatted. But, if you don't have individual files, or databases that are large, you might want to consider one of our controllers that can divide our 65 megabyate (formatted) hard disk into two equal volumes of 32 megabytes each.

Reliability

We offer you a choice between iron oxide and plated media—the stuff that covers the hard disk and gives it its magnetic properties. Iron oxide is,—well, it's rust. If you inadvertently joust your disk, you may cause the low flying head to dig out some iron oxide. A little rust flake can ruin your whole day. Plated media is more resistant to damage, and if it happens, less data is lost.

We offer both types of hard disks. The iron oxide is older

technology, and quite frankly, manufacturers understand it better. Their better understanding, combined with some of the special head locking mechanisms, gives us peace of mind when we sell you one.

Power

Hard disks consume power. Our small, half-high hard disks consume so little power that you can use them with your existing IBM PC power supply. If you plan to use lots of slots, you'll want to increase your power supply to be safe. We offer the same amount of power for your PC that comes in the XT.

Our Customers

Some folks just never feel comfortable buying mail order. They forget that Sears began as a mail order house or that IBM is now into mail order. But, if it helps, here is a *partial* list of customers who have felt comfortable to buy from us.

IBM	Sears
American Express	Honeywell
U.S. Army	MIT
AT&T (Bell Labs)	RCA
Bausch & Lomb	Lockheed
Xerox	Sperry

Easy to Install

If you're like most of us, raised on the boob tube rather than the Great Books, you'd rather see the movie than read the book. Well, now you can choose to read our installation manual or for only $9.95 more, you can get a VHS or Beta video cassette showing the simple steps for installation.

Warranty

We offer you a one year warranty on our hard disks—the same as IBM on the AT and 90 days on the tape drives. (It's all the manufacturer gives us.) If anything goes wrong with your tape or disk drive or hard disk, send it back in the box it came in. However, we have found that we can usually solve the problem over the phone. So call first for a return authorization number because we can't accept any returns without it.





Complete Hard Disk Kits

Formatted MB	Height	Plated Media	Average Access	Transfer Rate	PC or PC/XT	AT
10	½	no	85 msec	5 Mbits/s	$ 395	$ N/A
10	½	yes	85 msec	5 Mbits/s	$ 495	$ N/A
21	½	yes	85 msec	5 Mbits/s	$ 795	$ 595
21	Full	no	30 msec	5 Mbits/s	$ 1,535	$ 1,340
32	½	yes	85 msec	5 Mbits/s	$ 995	$ 795
32	Full	no	30 msec	5 Mbits/s	$ 1,775	$ 1,575
65	Full	no	30 msec	5 Mbits/s	$ 2,295	$ 2,070
100	Full	yes	18 msec	10 Mbits/s	$ 4,995	$ 4,995
Removable Hard Disk						
10	½	no	90 msec	5 Mbits/s	$ 1,095	N/A

Tape Systems and Subsystems

Formatted Storage Capacity	Height	Data Transfer Rate (k/sec)	PC or PC/XT	AT
60 Mbytes	½	88	$ 995	$ 995
60 Mbytes Subsystem		88	$ 1,295	$ 1,295
21 Mbytes (unformatted) Start/stop Subsystem		24	$ 595	$ 595
26 Mbytes Floppy Tape® Subsystem		31	$ 749	$ 749

Controllers

All of our hard disk and tape controllers are available separately: Please call for prices.

Subsystem Chassis

Any of our disk or tape units are available in an external subsystem for an additional $250.00. You can mix & match any of our ½ high hard disks or tape drives together or add any single full height hard disk.

Tape Cartridges

Express Certified 555 foot 310 Hci ¼-inch Data Cartridge **$35.00**

Power Supply

130 Watt Power supply **$75.00***
150 Watt Power supply **$125.00**

*with the purchase of any drive

Comes complete

All Express Systems products come complete with the appropriate software, tape and/or hard disk controllers, and cables where required. Hard disks are formatted and tested with the PC DOS of your choice. All drive sizes are formatted capacities.

If your application requires a stacking kit, power splitter cables, daisy chain cable, or some other variation, we'll supply these items at a nominal charge. We even ship our hard disks with Command Assist™ an on-line DOS-like manual to give you help with your DOS commands.

More questions?

Because we spend so much attention on the front end with ensuring that our disks will arrive in working order, we have a customer service department that, unlike many of our competitors, has little to do. When you need us, you won't get a constant busy signal.

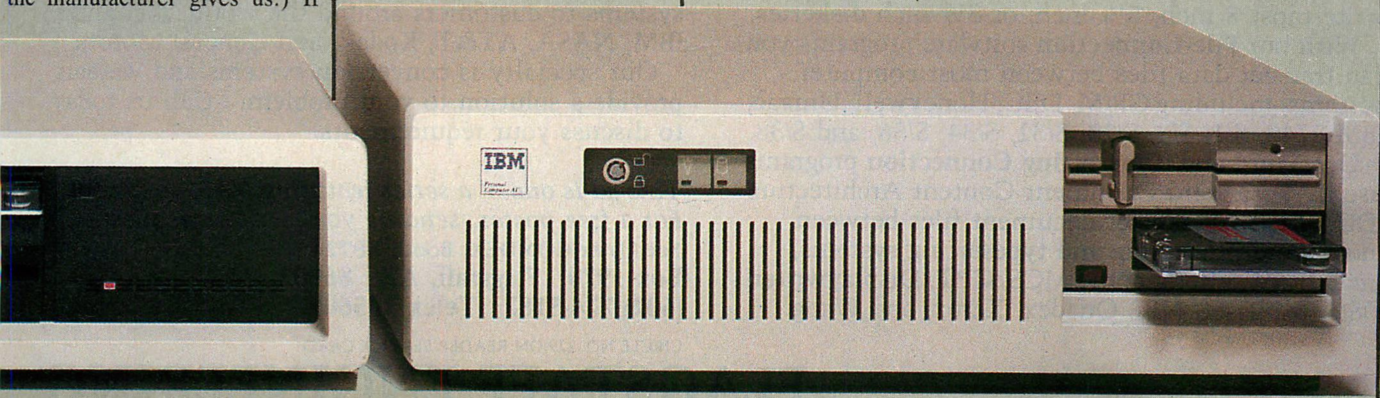
Call our friendly, knowledgable customer service staff to get answers to your questions—before or after the sale. Our people, who know the PC, can talk you through the sticky parts, and they'll respond to you quickly. Just call us.

How to order

Pick up the telephone and call 1-800-341-7549, to order. We accept Master Card, VISA, American Express and Diners Club. Or send a cashier's check or money order (We'll take a check, but you'll have to wait for it to clear) and tell us if you want one of our recommended configurations or you want to mix and match yourself. Corporations with a DUNS number may send purchase orders for quantities over five.

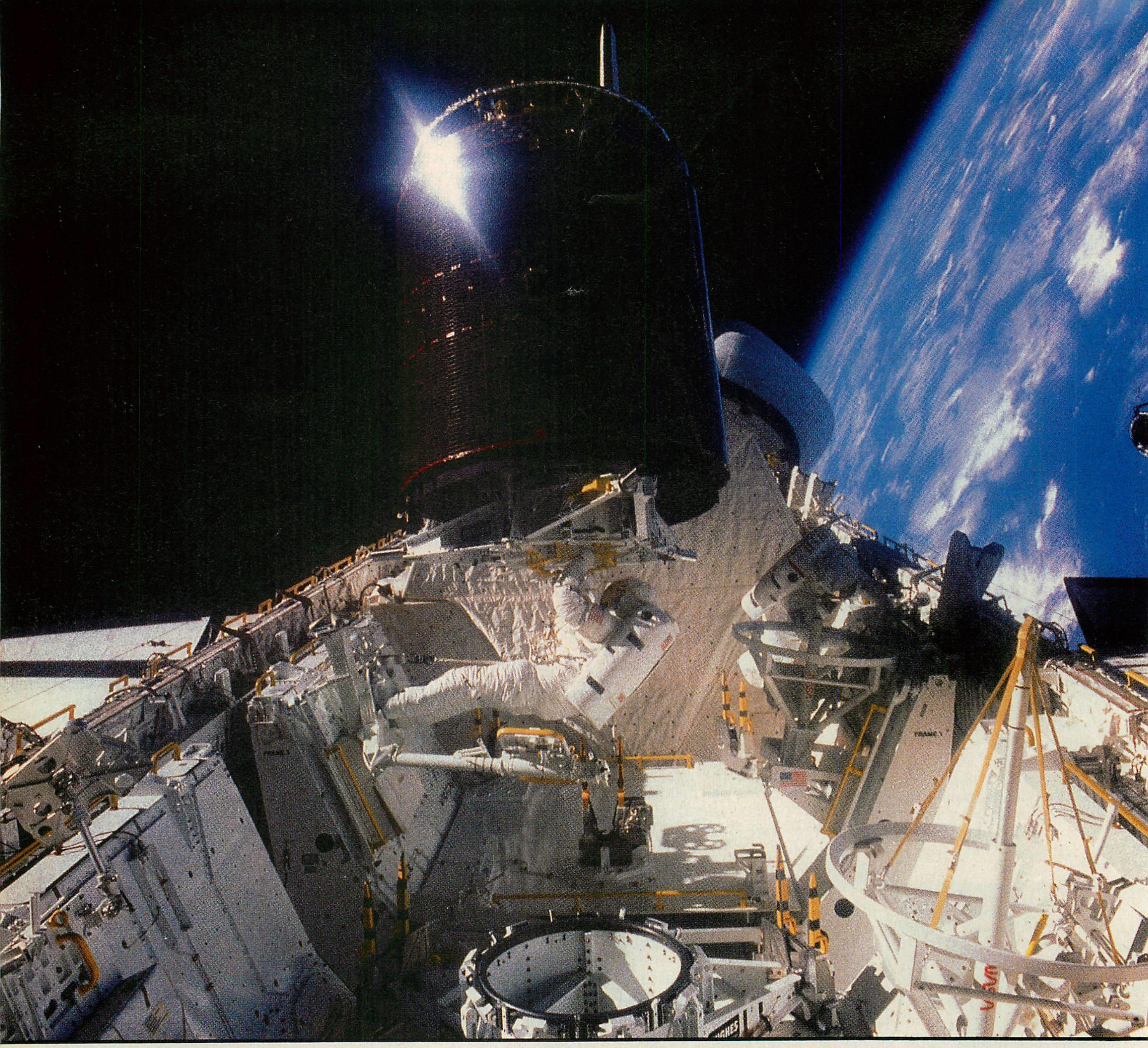
EXPRESS SYSTEMS

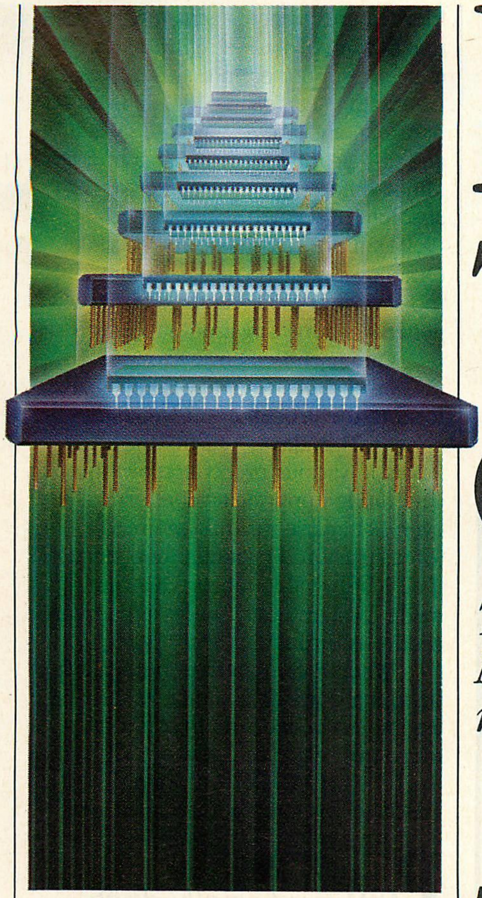
Call Toll Free 1-800-341-7549 Ext. 1300
In Illinois call (312) 882-7733 Ext. 1300
Express Systems, Inc., 1254 Remington, Schaumburg, IL 60195



Command Assist is a trademark of Micro Design International

CIRCLE NO. 240 ON READER SERVICE CARD





PASM, TASM, and CHASM

Three assemblers challenge Microsoft's Macro Assembler; MASM retains its preeminence.

TED MIRECKI

The IBM/Microsoft Macro Assembler (affectionately known as MASM) has been an inviting target for criticism ever since its introduction in 1981. Its complexity, slow speed, and numerous bugs have drawn potshots from users and from software companies who have tried to improve on the assembler's sorry performance. The latest salvo being directed at MASM comes from three new assemblers: Phoenix Software's macro assembler (PASM), TURBO EDITASM (TASM) from Speedware Inc., and the Cheap Assembler (CHASM), a user-supported product developed by Whitman Software.

These assemblers miss their target by a good mile—and not only because, in its latest incarnation, MASM is much improved over the original (for a review of the most recent version of MASM, see "Same Language, New Architecture," Ted Mirecki, October 1985, p. 48). The new assemblers miss simply because they do not do a reliable job of assembling valid source programs. Features of the three assemblers are compared to MASM's in table 1.

This review judges PASM, TASM, and CHASM against the current standard, Microsoft's MASM 3.01. To test the assemblers, the same six programs used originally to test MASM again were pressed into service (see table 2). All have been previously published by *PC Tech Journal* and all assembled cleanly with MASM and executed properly. The test environment was also the same: an IBM PC with 640KB of memory, running under DOS 3.1. The assemblers were run from a RAM disk, with source file input from a floppy drive and object file output to the RAM disk.

CLOSEST IN CONCEPT

The Phoenix assembler is the closest in concept to MASM. It operates in much the same fashion, can assemble the same source files with no changes, and produces standard .OBJ files for linking with either DOS LINK or Phoenix's Plink. Modules produced with PASM, MASM, or high-level languages may be mixed in the same link step.

PASM's user interface is UNIX-like. Assembler options are preceded by dashes, and the name of the input file must be placed last on the command line. Unlike the DOS interface, however, PASM does not allow interactive entry of file names or options if these are not typed on the command line. All missing options and output file names are defaulted, and if the input file is not specified, the user is returned to DOS.

One useful feature of the command line interface is the ability to specify symbol definitions at assembly time. The assembler directive -dNAME=VALUE is equivalent to the

ILLUSTRATION • ALAN HASHIMOTO

source statement NAME EQU VALUE and can control conditional assembly without changing the source file.

Several bugs are apparent in PASM's handling of the command line parameters. Preceding the input file name with a directory path or drive identifier sometimes results in the message "Cannot open input file." Similarly, the object file sometimes cannot be opened if drive or directory information is included as part of its name. Occasionally, the object file is just not created, and no error message occurs.

The listing produced by PASM is similar to MASM's, except that the symbol table is suppressed by default and must be specifically requested. Cross-referencing of symbols is not provided. The manual describes two directives, .CREF and .XCREF, that supposedly turn on and off the generation of cross-reference information, but where this information goes and how to display or print it could not be discovered.

The source language syntax accepted by PASM is almost identical to that of MASM. All of the major assembler features (macros, structures, conditional assembly, value returning operators) are supported in the same fashion. Instruction sets for the 186, 286 (in both real and protected modes), 8087, and 287 processors are fully supported. Minor differences exist in the way that absolute offsets are specified and in PASM's more rigorous error checking. On occasion, the error checking seems too severe; some acceptable constructs generate spurious warning messages.

One useful extension to MASM syntax is the ability to define labels that are local to the procedure in which they appear. This allows reusing the same labels in various procedures without generating duplicate symbol errors. An even better use for this feature is to enforce structured programming practices: if all labels within a procedure are made local, the only way to enter a procedure is through its entry point.

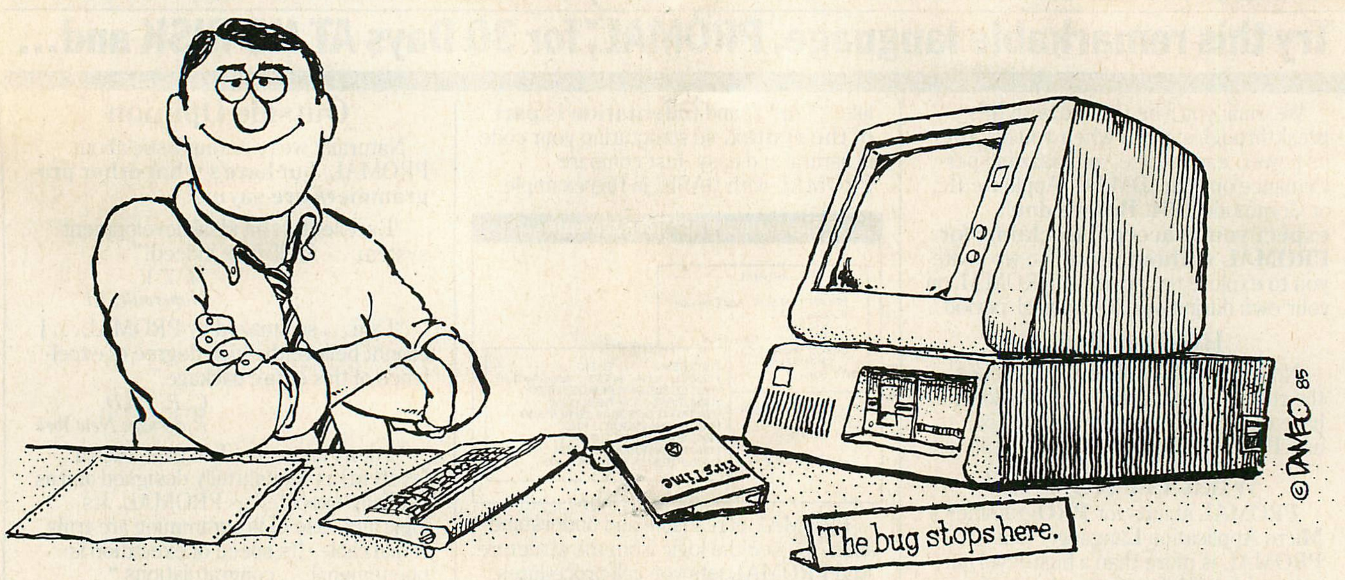
When PASM operates properly, it is lightning quick. It is the fastest of all the assemblers tested, and in some cases, more than twice as fast as MASM (see the time for VDISK). Unfortunately, the speed seems to be at the expense of

PASM is especially disappointing given Phoenix's good reputation in other systems software. Somehow, the company seems to have released a pre-beta test version as a finished product.

accuracy: it produced clean assemblies on only two of the six tests.

The worst failure occured on the PACKDIR program; the system locked up early in the first assembly pass and needed a keyboard reset to recover. The macro test program with ten iterations could not be assembled because PASM ran out of conditional stack space and gave up. The shortest program, BOOT, generated no error messages, but the resulting object file could not be linked because of segment fixup overflow. Such errors are usually caused by near references to addresses in another segment, but the code in the BOOT program is innocent of such transgressions. The mildest assembly errors occurred on the second-largest

The First Idea-Processor For Programmers.



FirsTime™

Has features no other editor has.

- Fast program entry through single keystroke statement generators.
- Fast editing through syntax oriented cursor movements.
- Dramatically reduced debugging time through immediate syntax checking.
- Fast development through unique programmer oriented features.
- Automatic program formatter.

FirsTime is a True Syntax Directed Editor.

As the world's most advanced syntax-directed editor, FirsTime lets you work with ideas by taking care of the low-level syntax details of your program. For example, you can generate complete statement skeletons with one keystroke. Move the cursor from one procedure to the next with one keystroke. Type in code, and it's instantly formatted (you specify the rules). Type an error, and FirsTime warns you immediately. You can continue working if you wish. Later, you can use the search-for-error command to find the error and fix it.

FirsTime Has Thorough Error Checking.

FirsTime not only checks your syntax, but also semantics. FirsTime identifies:

- Undefined variables, types and constants.
- Assignment statements with type mismatches.
- Errors in include files and macro expansions.

To Order Call: (201) 741-8188 **or write:**

SPRUCE TECHNOLOGY CORPORATION

P.O. Box 7948
Shrewsbury, NJ 07701

FirsTime Lets You Work With Ideas.

The *Zoom command* gives you a top down view of your program logic.

The *View macro command* shows the expansions of a C macro while in the editor.

The *View include file command* instantly shows you the contents of an include file.

The *Transform command* allows you to change a statement to another similar statement, for example, a *FOR* to an equivalent *WHILE*.

The *Search for next error* command allows you to find errors throughout your program.

The *Cursor movement commands* let you traverse your program by logical elements, not just characters.

FirsTime Works With Existing Files.

FirsTime works with standard ASCII files, so you can edit any existing source files.

FirsTime for Turbo Pascal	**$ 74.95**
FirsTime for dBase III	**$125.00**
FirsTime for MS–Pascal	**$245.00**
FirsTime for C	**$295.00**

In Germany, Austria and Switzerland contact:
Markt & Technik Software Verlag
Munchen, W. Germany
(089) 4613-0

program, PLOT, in which only two spurious warnings were generated.

The two clean assemblies were of VDISK and MACTEST with only one iteration. PASM's speed on these is truly amazing, but considering its sorry performance on the four others, its speed is of academic interest only.

The documentation, in a standard 8½-by-5½-inch binder, includes a complete instruction set reference but not a hardware reference. The manual does not presume to teach the subtleties of the 8086 family of processors; for this the user is referred to Intel documentation (a list of titles is provided in an appendix). Although not as complete as that for IBM's version of MASM, the PASM manual is superior to Microsoft's. The package includes no files besides the assembler itself. Several example files are mentioned in the documentation, but none was present on the disk.

PASM is especially disappointing given Phoenix's good reputation in other systems software. Somehow, the company seems to have released a pre-beta test version as a finished product. It shows good conceptual design, maintaining adequate compatibility with MASM and providing several useful extensions. Its execution at this stage is totally unacceptable, however.

A LEARNING VEHICLE

Speedware's TURBO EDITASM is designed to operate much like Borland International's Turbo Pascal. It includes an integrated editor that can be reentered on assembly errors, and its output can be directed either to a .COM file or to memory for immediate execution. The base version provides only these two choices. An enhanced version at double the price claims to produce .OBJ files and can be run in noninteractive fashion from DOS batch files.

When the interactive version of TASM is invoked, it presents a main menu with the functions of loading and saving files, entering the editor, assembling, executing, and controlling assembler and listing options. This user interface works satisfactorily, but it suffers from one weakness: when saving, the name of the current source file is not presented as the default; it must be typed in by the user. Although the file name is displayed above the menu, the possibility still exists for the user to enter the wrong name, with potentially disastrous results.

The editor follows the ubiquitous WordStar conventions. Common cursor operations are provided by the PC's keypad. An installation program may be used to substitute more mnemonic key sequences for WordStar's cryptic Ctrl-letter combinations.

The syntax accepted by TASM is fairly close to MASM, at least for programs assembled to an object file. Instructions for the 8088, 186, 286 (real mode only) and the 8087/287 are supported. Major advanced features, such as macros, structures, and conditional assembly are implemented, including the repeat pseudo-ops REPT and IRP, but not IRPC. Some seemingly minor differences can turn out to be major nuisances in assembling MASM programs: TASM does not support the COMMENT directive, does not allow any key words (instruction mnemonics or operators) to be used as symbols, and complains about extra pairs of parentheses in expressions.

Programs that are to be assembled directly to a .COM file rather than a .OBJ file are different: they must not contain any segment definitions or ASSUME statements. The rationale for this is that a .COM file contains only one segment, so it does not need a name.

This prevents the use of segments defined at an address, a scheme often used to address data whose absolute location is fixed (for example, the BIOS data area at segment 40H).

TASM's incompatibilities are minor when developing programs from scratch, but can be major stumbling blocks when assembling programs written in MASM syntax and when incorporating MASM subroutines in TASM programs. When assembling to a .COM file, TASM does not support external symbols, so subroutines must be included in source form at the assembly step, not as object files at link time. They must be converted into TASM syntax, a task that may be far from trivial. Removing references to absolute segments may require changes to program logic as well as to assembler directives.

Apart from this, the interactive operations of TASM, at least for creating simpler .COM files, are good. Because TASM is written in assembly language, it is fast: it beats MASM by a mile, considering that TASM can create an executable file in less time than MASM takes for the assembly step alone.

Assembler options are set on a sub-menu and remain in effect until changed (they need not be respecified at each assembly). Both .COM and .OBJ output may be turned off, allowing very quick assemblies for error-checking only. Alternatively, executable code may be assembled directly into memory for immediate execution without subsequent loading. At the user's option, an assembly error may either display a message and continue or cause a return to the editor, with the cursor positioned at the offending line of the source program. Because both the editor and assembler are in memory, no time is lost reloading. The listing, which is essentially the same as MASM's, may be directed to the screen, printer, or suppressed entirely. Another option turns on the cross-reference facility, which creates a table in memory for subsequent display. Unlike MASM's separate CREF step, this cross-reference is produced concurrently with the assembly.

TASM has difficulty generating object files, and its performance on complex macros is not very reliable.

TASM's main menu also provides a calculator option that does arithmetic in decimal, hex, octal, and binary. It can evaluate any legal expression in an assembly language statement, using bit manipulation, comparison and type operators, and arithmetic operations. The expressions may contain symbols from the program last assembled, because the symbol table is saved after each assembly—a useful feature.

TASM has difficulty generating object files, and its performance on complex macros is not very reliable. Both the interactive and the batch versions created a valid object file for only one of the test programs, PLOT (see table 2). For BOOT, TASM generated five phase errors (differences in generated code between the two assembly passes); and for VDISK, one message about a relative jump out of range (more than 127 bytes distant). No error messages were generated by the assembler or linker for PACKDIR, but address offsets in the executable code were from the start of the data segment, not from the start of the group.

The creation of .COM files directly from these same programs was not tested because the conversions necessary to get them into the required TASM syntax were too extensive. However, shorter programs incorporating instruction sequences similar to the ones causing these errors assembled cleanly. Therefore, whether TASM fails in generating object code or in assembling larger programs is not clear. The former is the more likely, given the errors on the short BOOT program.

Although TASM handles simple macros well, it gets confused on the complex, nested macros of the MACTEST program (see the MACTEST.ASM listing on p. 59 in the October issue; it is also available for downloading on PCTECHline). It generates an avalanche of error messages, many of the uninformative "syntax error" variety, and most of them for statements that are clearly documented as valid. The same constructs cause no problems when in simpler contexts.

TASM's documentation is concise yet fairly complete, containing instructions for using the main menu and the editor as well as an adequate instruction set reference. It includes a list of reference books, both tutorial and advanced, and even a list of periodicals of interest to assembly language programmers. The manual consists of some 250 8½-by-11-inch pages bound along one edge, meaning that it will not stay open and is too large to prop up on the key-

TABLE 1: *Features Comparison*

	MASM	PASM	TASM	CHASM
Version	3.01	1.0	1.03B	4.0
Assembler size	77.4KB	101.1KB	60.4KB	52.9KB
Requirement	128KB	192KB	128KB	128KB
Creates .OBJ files	Yes	Yes	Full ver. option	No
Creates .COM files	No	No	Full. ver. option	Yes
Cross-reference	Separate step	No	Yes	No
Other files	Link, DEBUG, LIB, MAKE, CREF	None	Integrated editor, calculator, examples	Documentation, primer, examples

The three challengers to MASM have not matched MASM's features and abilities in their current versions. None can be recommended as a full replacement for MASM.

TABLE 2: *Performance Comparison*

	SOURCE SIZE	MASM	PASM	TASM
BOOT.ASM	4.KB, 140 lines	0:10	0:05[a]	0:08[b]
PLOT.ASM	7KB, 280 lines	0:15	0:08[c]	0:09
PACKDIR. ASM	20KB, 480 lines	0:37	crashed	0:20[d]
VDISK.ASM	74KB, 2,150 lines	2:16	0:50	1:09[e]
MACTEST.ASM	N/A			
1 iteration		0:15	0:08	[f]
10 iterations		1:29	[g]	[f]

Assembly times in minutes:seconds
[a]*Segment fixup overflow on link*
[b]*Five spurious phase errors*
[c]*Two spurious warnings*
[d]*Data references not relocated correctly*
[e]*Spurious error: conditional jump out of range*
[f]*Generated too many errors; would not assemble*
[g]*Exceeded conditional stack space*

Only programs without footnotes assembled and linked cleanly. Speed comparisons on the others are not very meaningful because of errors. CHASM could not be tested on these programs because of different syntax rules.

board. The text makes many comparisons to MASM and assumes that the reader is familiar with the Microsoft product. This practice may puzzle the type of user for whom TASM is best suited: the assembly language novice.

The bottom line on TASM is that it is a convenient, speedy, interactive assembler for uncomplicated .COM programs. Its base version is especially suited for assembly language novices as a learning vehicle until they graduate to a full-featured assembler, and for users of MASM as a second, small assembler to replace DEBUG's mini-assembler for small utility programs for which an editor, MASM, and LINK are too unwieldy.

As a tool for developing larger applications, however, TASM has a long way to go before it can be considered as a replacement for MASM. The creation of object files is a hit-or-miss proposition, macros are not quite fully implemented, and their expansions are less than reliable. Until these problems are corrected, the .OBJ version cannot be recommended as a usable product.

FARTHEST REMOVED

Whitman Software's Cheap Assembler is a user-supported program. The author requests a contribution if the user finds the program useful. Only the base version, which does not implement macros and structures, is to be distributed free of charge, however. A $40 registration fee entitles the user to the full version that includes these features.

CHASM operates on ASCII source files produced by any external editor, and produces .COM files directly. It has no object file option. It is the farthest removed from MASM, using a source language syntax that differs significantly. Programs written for MASM, PASM, and TASM require significant revisions

before they will assemble with CHASM. The reason for this is that Whitman was guided by *The 8086 Book* by Russell Rector and George Alexy (see "Studying Assembly Language," November 1985, p. 189), which does not describe any actual assembler implementation.

Like TASM, CHASM does not allow segment definitions, ASSUME statements, or external declarations in programs. However, the differences from the MASM standard go beyond this. If

For experienced programmers, CHASM's limited capabilities (its inability to create .OBJ or .EXE files as well as its incompletely implemented macros) place it in the toy program category.

the ENDP (end of procedure) statement is labeled with the procedure name, a duplicate definition error results. The PTR and SHORT operators are not supported; instead, *B, W,* and *S* suffixes are added to the instructions. Furthermore, the OFFSET operator is implemented as a function, meaning that its operand must be enclosed in parentheses, and the DUP operator for data definition statements is not implemented; a DS (define storage) statement provides a similar capability. The user may discover more differences. Even Whitman is not aware of all of them; he admits that he does not even own a copy of "that other assembler."

CHASM does support structures, records, conditional assembly, and a subset of MASM's macro capability. As a result of its oddball syntax, it could not be tested on any of the programs used for the other assemblers without a total rewrite of each one. It seems to handle simple .COM programs as well as TASM, but is not as fast. On programs of similar size, CHASM's assembly times are longer than the three-step process of MASM, LINK, and EXE2BIN. The current version of CHASM is written in Turbo Pascal; previous versions were in BASIC. Whitman promises a future version written in CHASM's own language.

The documentation follows the standard practice for user-supported software: a text file on disk to be

printed by the user. This 80-page document adequately covers the operation of the assembler and its syntax; users are referred to the Rector and Alexy book for information on the instruction set and the microprocessor structure. In fact, that book is stated to be a requirement for using CHASM. Included with the assembler are several example programs that provide useful utilities: changing volume labels and file attributes, word counting, etc. The most useful inclusion is another print file containing tutorial material for users totally new to assembly language.

Novice programmers often begin by assembling programs published in books and magazines, and these are usually written for MASM. Translating them to CHASM requires a better knowledge of both assemblers than the novice can be expected to possess. For experienced programmers, CHASM's limitations (inability to create .OBJ or .EXE files, incompletely implemented macros) place it in the toy program category. It cannot be recommended.

The competitors are numerous but Microsoft's MASM still reigns supreme. Other assemblers simply are not reliable. Though noble in conception, Phoenix's big PASM assembler is crippled by bugs; a later version may challenge MASM. Whitman Software's CHASM is not powerful, not standard, and, in the final analysis, not useful. New users looking for a small assembler will do better with the base version of Speedware's TASM.

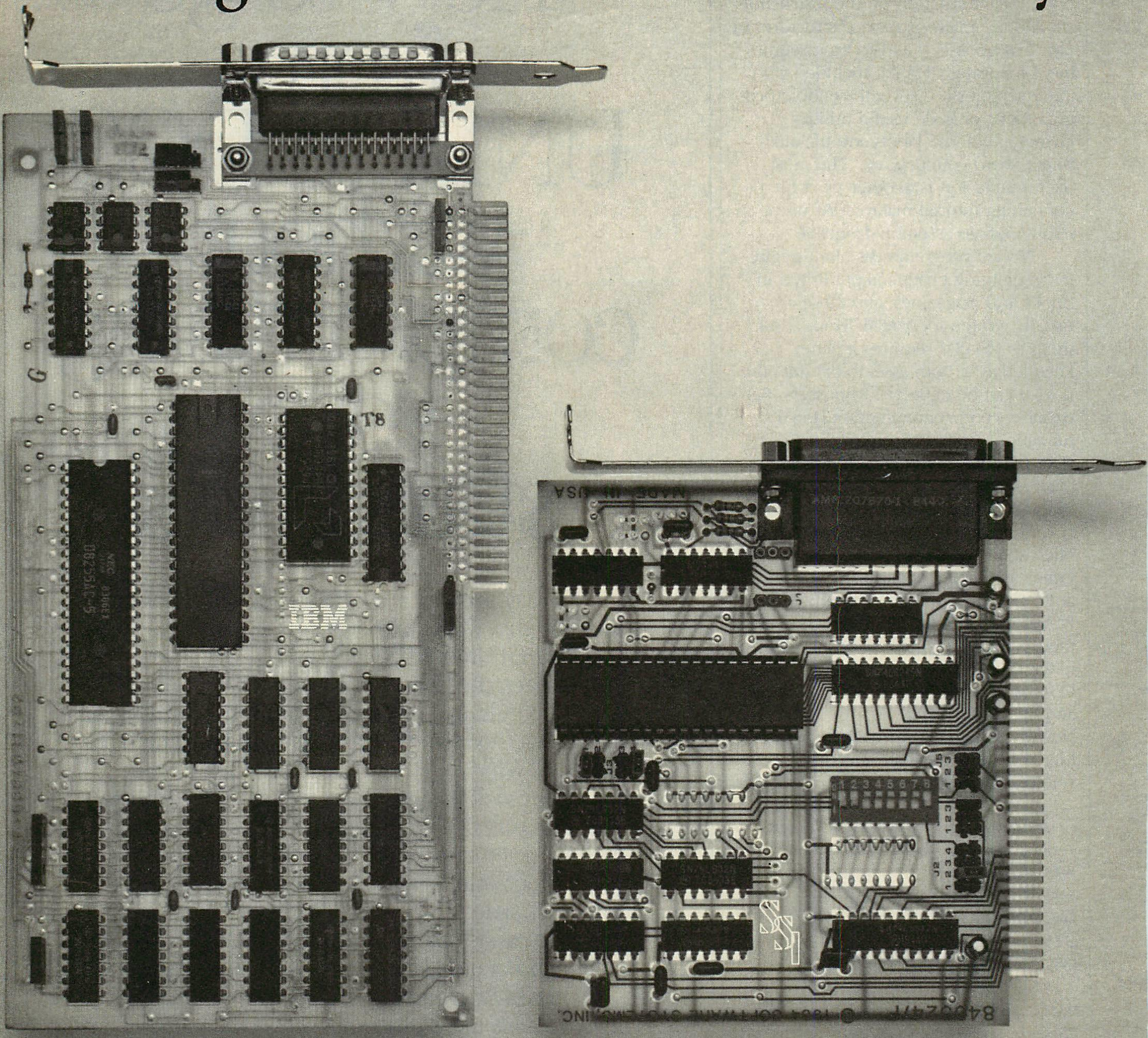
CHASM: voluntary contribution (base version); $40 (full version)
Whitman Software
136 Wellington Terrace
Lansdale, PA 19446
215/362-8526
CIRCLE 342 ON READER SERVICE CARD

PASM: $395
Phoenix Software Associates
320 Norwood Park South
Norwood, MA 02062
800/344-7200
CIRCLE 340 ON READER SERVICE CARD

TASM: $49 (base version); $99 (full version)
Speedware Inc.
118 Buck Circle, Box T
Folsom, CA 95630
916/988-7426
CIRCLE 341 ON READER SERVICE CARD

Ted Mirecki is a corporate planner responsible for developing decision support systems on a variety of hardware. He has a master's degree in computer science.

Choosing a remote 5251 emulator is easy:



The biggest or the best.

The fastest, smallest and most efficient way to connect your PC to the System 34/36/38.

When you need to work with system data, you want an emulator that's up to speed. At 19,200 bps, the PC 5251/PLUS™ from Emerald Technology transfers files faster than any other Model 12 emulator—it's twice as fast as a standard 5251 terminal!

The PC 5251/PLUS was developed by Software Systems, Inc., experts in PC-System/3X communications. PLUS is streamlined to require ⅓ the memory of IBM's emulator and still offer all these features:

Hot Key—Lets you switch from PC to emulator mode and back with the stroke of a key.

Keystroke Files—Lets you predefine responses to system prompts for automatic or unattended operation.

Small Size—Lets you use the short slots on an IBM PC, XT, AT or compatibles.

Light Pen Support—An alternative to keyboard input for special applications.

File Transfer Support—Works with SSI's Emulator Transfer Utility™ for high speed bidirectional file transfer.

Backed by extensive technical support from the System/3X specialists, Emerald Technology's PC 5251/PLUS is not the biggest, but the best. Call today for a dealer near you.
(206) 462-8200

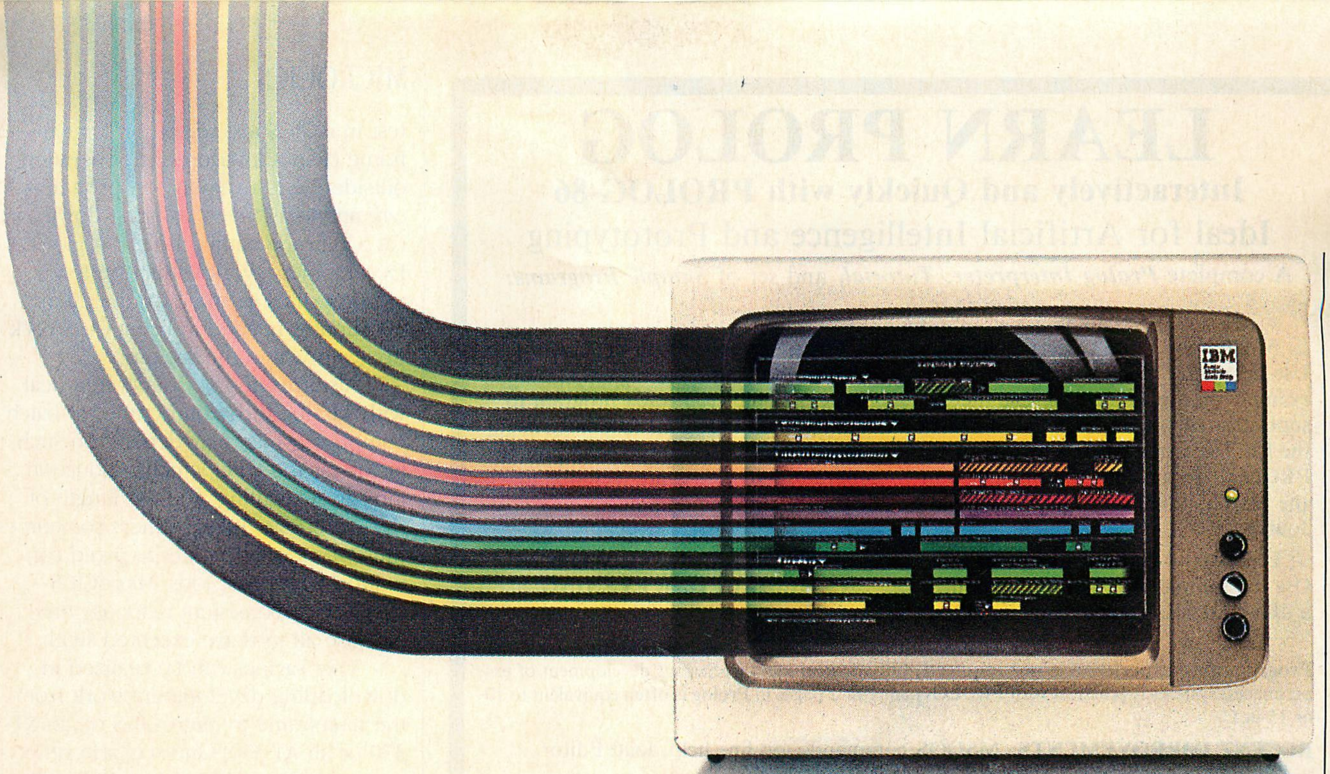
The best solutions. The best support.

Emerald
Technology Group Inc.

1601 116th Ave. NE, Bellevue, WA 98004 **(206) 462-8200**

PC/5251 PLUS and Emulator Transfer Utility are registered trademarks of Software Systems, Inc. IBM is a registered trademark of International Business Machines.

CIRCLE NO. 234 ON READER SERVICE CARD



Personal CICS

Unicorn Systems' MicroCICS provides a CICS development system on a desktop.

BRANDY C. DE SHAZO

Programmers who are familiar with CICS (customer information control system) applications have long wished for a personal CICS environment. Developing such applications has always been a dubious undertaking, usually plagued by coordinative traffic jams. Long lead times for definition of the CICS environment and frequent interruptions that make CICS unavailable are among the drawbacks.

A lot of lead time is spent interfacing with other groups that are responsible for various pieces of the CICS puzzle. A CICS project may require coordinating the efforts of systems programmers for CICS table updates, facilities management for space allocations, database administration for file creation, and the operations staff for scheduling. Although each request is not time-consuming in itself, the amount of time required seems to increase exponentially by the number of persons involved. Often, valuable test time is wasted waiting for various tasks to be completed.

Experienced CICS designers estimate the time needed to complete the actual work, than add extra time for the unknowns—those times when the test system is not available or when it might take longer than anticipated for a CICS system table to be updated correctly or when programs or test files are destroyed accidentally by someone's test application. The accuracy of time estimations for unknowns is crucial to meeting project deadlines, but they are difficult to get right in these situations.

A new and powerful development and testing tool called MicroCICS, by Unicorn Systems, solves these problems by providing a sheltered, personal CICS system that fits on a desktop. MicroCICS supports the entry, compilation, testing, and debugging of CICS applications programs on IBM PC-370 hardware using IBM's VM/PC operating system. The PC-370 hardware is actually a set of two option boards that can reside in either a PC/XT or a PC/AT chassis. (The board set configuration for the XT is subtly different from that for the AT, basically in the way memory is managed.)

With MicroCICS, a programmer is able to generate BMS (basic mapping support) maps, to define, load, and edit VSAM (virtual sequential access method) files, and to modify system table information. Existing CICS programs, mapsets (binary data sets containing CICS maps), and test data can be downloaded from the mainframe, modified, then uploaded when the required changes have been completed.

The premier advantage of MicroCICS is its ability to isolate new users from programmers testing new CICS programs. Although programs under

ILLUSTRATION • BOB AIESE

test in a CICS test region on a mainframe cannot interfere with operations outside the test region, test programs *can* and sometimes do bring down CICS running within the test region. Development work within the test region grinds to a halt for the 20 or 30 minutes needed to bring CICS back up again within the region.

Also, mainframe users often deal with varying turnaround time for batch compiles, which may depend on such factors as the date, which production jobs are executing, and the length of the job queue. Programmers sometimes must work at odd times to avoid contention with other jobs. MicroCICS users enjoy consistent response time because all jobs are executed singly.

Cost savings can be realized in downloading development work from the mainframe to MicroCICS on an XT-370 or an AT-370. One scenario reflects an actual MicroCICS customer installation and involves two alternatives: a 4300 mainframe dedicated to CICS development by 10 programmers and a lab of 10 AT-370 machines each running MicroCICS. The cost of the 4300 system and software is about $300,000, not including environmental support (air conditioning, raised floors, and so on) or system support programmers. The cost of the 10 desktop machines and software comes to a little over $150,000.

The savings would be less in situations where programmers simply were moved from a mainframe to several desktop installations of MicroCICS without eliminating an actual mainframe machine. As COBOL and assembly language work also can be downloaded to the desktop machines, some creative reorganization of a department's development efforts might garner considerable savings even if downloading CICS development alone might not justify the effort. In any case, improving response time pays high dividends in programmer productivity, as studies by IBM in Canada have shown.

THE BASICS

MicroCICS's minimum hardware requirements are an IBM XT-370 or AT-370, 640KB of memory, one fixed-disk drive, and a monochrome or color monitor. MicroCICS requires no modification to the system hardware or operating system software, and once installed it will not interfere with machine operation when not executing.

Currently owned XTs and ATs can be upgraded by installing the XT-370 or AT-370 option kits. Both 370 models were used in evaluating MicroCICS for

TABLE 1: *MicroCICS Performance Comparison*

SOURCE LINES	AT-370	XT-370
500	2:25	3:50
1,000	3:50	6:00
1,500	5:45	8:45
2,000	6:40	10:40
2,500	8:35	13:45
3,000	10:00	16:35

Above are compilation times (in minutes: seconds) for COBOL programs running under the MicroCICS environment. The AT-370 is faster by a considerable margin.

TABLE 2: *CMS Commands*

ACCESS	ESTATEW	IDENTIFY	RELEASE
CMDCALL	EXEC	LISTFILE	RENAME
COMPARE	EXECIO	MAKEBUF	SENTRIES
DEBUG	FILEDEF	NUCXDROP	STATE
DEFAULTS	FINIS	NUCXLOAD	STATEW
DROPBUF	GENDIRT	NUCXMAP	SYNONYM
ERASE	GLOBAL	PRINT	TYPE
ESTATE	GLOBALV	QUERY	XEDIT

These commands do not conflict with MicroCICS operation. Other CMS commands require memory used by MicroCICS and must be executed from CMS itself.

this article. Although response time certainly was acceptable using the XT, performance improved noticeably on the AT, as expected. This was true particularly for disk-bound tasks such as program compilations, which are not performed by MicroCICS software; instead these are processed by a separate COBOL compiler using the 370 instruction set. Table 1 lists compilation times for various COBOL programs; it is provided simply as a basis for comparison of XT versus AT performance.

Although modems are not required for MicroCICS processing, they will be necessary to transfer data or programs between a mainframe and VM/PC. If several MicroCICS systems are used for development purposes, only one machine intended for host communications needs to be equipped with a modem. Disks can be used to transfer data from the host-linked system to the others.

Any hardware that normally would be required for using 3278/79 terminals over telephone lines (such as a 3274 controller) also is required for use of the PC-370 hook-up over telephone lines. Alternatively, any of the communications hardware provided by third party suppliers may be used in PC-DOS mode to communicate with the host.

The minimum software requirements for MicroCICS are the IBM VM/PC operating system (which includes the text editor XEDIT and mainframe link program VMPCSERV), the IBM OS/VS COBOL compiler, or a native 370 assembler such as Assembler H. Installations running VM/CMS (virtual memory/conversational monitor system) on a mainframe can use VMPCSERV, which controls file communications with a VM/CMS mainframe and is included with the VM/PC operating system to provide file exchange. It also allows the use of mainframe CMS minidisks as local minidisks and mainframe VM printers as VM/PC printers.

For installations running OS/MVS with TSO, the IBM product TSOSERV (which can be leased or purchased from IBM), is required for establishing the micro-to-mainframe link through VM. Other telecommunications products that support data transfer with the mainframe also can be used, although not as easily or as transparently. Like VMPCSERV, TSOSERV furnishes file transfer between the mainframe and PC, and allows the execution of TSO commands from a local VM/PC session and the spooling of files to the mainframe for printing. TSOSERV permits VM/PC to access mainframe sequential and partitioned data sets in the same manner they would access CMS files.

MAINFRAME COMPATIBILITY

MicroCICS provides near full compatibility with releases 1.5 and 1.6 of IBM CICS. Support for CICS release 1.7 is currently being tested. MicroCICS emulates one release at a time; the emu-

lated version must be selected when MicroCICS is invoked. The IBM VM/PC operating system used by the XT-370 and AT-370 provides CMS XEDIT, a text editor identical in function to the XEDIT used on CMS mainframes. For programmers who already are familiar with mainframe XEDIT, the MicroCICS learning curve can be very short. However, because many mainframe 370 programmers use other text editors (typically TSO/SPF and ISPF), this is not a widespread advantage.

Incompatibilities with mainframe CICS hinge upon mainframe peripherals and other physical considerations that cannot be duplicated in a desktop environment. *Any* CICS command valid in a mainframe CICS environment can be compiled by MicroCICS without error. Certain commands and options of certain commands will have no function when executed, but they will not generate an error in compilation or execution. (A program that depends upon an unsupported command or option will not function as expected.)

Most unsupported commands are related to multitasking, which is impossible under VM/PC. For example, the ENQ RESOURCE command (which requests use of a resource exclusive from other tasks) accomplishes nothing because only one task runs at any given time under MicroCICS. Also, support for display formats other than 25 by 80, which is available under mainframe CICS, is not available in MicroCICS.

The standard mainframe release IBM 370 OS/VS COBOL compiler and Assembler H product can be downloaded to the XT-370 or AT-370 (but require the purchase of a separate license from IBM) and run on the desktop machines without modification. The MicroCICS command translator is almost source-compatible with its mainframe counterpart and requires little training. MicroCICS also provides a VSAM simulator that supports mainframe VSAM functions under CICS, including alternate indexing through a PATH.

MicroCICS is not *object*-compatible with mainframe CICS. While the MicroCICS command translator recognizes the same source commands as its mainframe counterpart, the lists of CICS calls it produces are slightly different and can be executed only on the desktop machine. This was done in part to make MicroCICS faster and more efficient, and also to establish firmly that it was in no way "lifted" from mainframe CICS. (It has not always been productive to emulate IBM too closely.) Source programs that have been compiled, run, and tested as being correct and in compliance with specifications can be uploaded to the mainframe and recompiled by the CICS processor. At that point, CICS features that had been nullified by MicroCICS (multitasking, for example) can be tested and modified.

MicroCICS displays descriptive menu screens that allow the user to select functions using PF (programmable function) keys. VM/PC maps the PF and PA keys onto Shift and Alt combinations with the PC's 10 function keys. Error messages are displayed in text on the screen to minimize reference checks. Much of the power of CMS is available from within MicroCICS: PF9 brings up a command prompt from which most CMS commands may be executed. (See table 2 for a list of PF9-available commands.) Those not listed in the table are excluded because they require memory areas also used by MicroCICS.

VM/PC's CP (command processor) commands are also available from within MicroCICS. They can be accessed either indirectly via CMS through the PF9 prompt or directly by pressing PA1 and bringing up CP's own command prompt. None of the CP commands conflicts with MicroCICS memory use.

Moving data back and forth between an XT-370 or AT-370 running MicroCICS and a 370 mainframe is handled transparently by the VM/PC operating system. With the VMPCSERV program installed and running on the host mainframe, no clear-cut functional boundary exists between the desktop machine and its remote host. Files are not explicitly uploaded or downloaded. If a file is resident on a mainframe disk device, MicroCICS can read that file as though it existed on a local disk drive; the user notices a difference only in access time. Once the read request is made to VM/PC, the physical location of the file is determined and VM/PC handles the movement of the data over the established physical link.

This is accomplished through the VM concept of minidisks, which are logical devices mapped onto physical storage media. While VM/PC cannot *define* minidisks on the mainframe, it can access any mainframe minidisk for which it has security clearance. Once clearance is granted, the mainframe minidisk can be read as though it existed on the desktop's own disk drives.

The performance of data transfer between desktop and host mainframe is dependent on the nature of the physical link, which is under the control of VM/PC. MicroCICS itself takes no part in data transfers after the request is issued

and does not affect transfer performance. One limitation of file transfer through VM/PC is that VSAM files may not be uploaded or downloaded in their actual unaltered form. VSAM files first must be converted to sequential files before transfer in either direction.

GETTING THE BUGS OUT

The MicroCICS debug processor is a sophisticated interactive system that goes beyond mainframe CICS's testing tools—the execution diagnostic facility (EDF) and the command interpreter (CI). It can be invoked in three ways:

1. The programmer can select Debug mode at the same time he initiates the execution of the program or transaction to be tested.
2. Once a transaction is in progress, pressing the PA3 key prior to a screen input will invoke the debugging facility. This will save time when only a portion of a multiscreen transaction needs to be tested.
3. Whenever a critical error occurs, the debug processor is automatically invoked by MicroCICS, giving the programmer the opportunity to analyze and correct the problem.

The debug processor furnishes a split-screen display providing a source code window at the top and a data item display area below. The programmer can see the source code as it is being executed, which eliminates the need to work from a hardcopy listing that may not be current. It also saves the time and paper required to print new listings for each coding change.

The source code window acts as a CICS source animator, showing 10 lines of the program source and highlighting the line currently being executed. The programmer may scroll forward and backward through the source, select a specific line number to be displayed, or search the source for the occurrence of a particular character string.

When executing a CICS command, the data item display area automatically shows the parameters being passed to CICS in hex and character formats, along with their storage locations. The parameters can be verified and modified. The programmer can specify the symbolic name of any program variable, which then can be viewed in the data item display area. The variable's data are shown in hex and character formats and may be modified. Numeric values can be entered as decimal numbers that MicroCICS converts automatically. All main storage locations can be displayed, and program data areas and fields can be modified by symbolic name.

The programmer can choose a specific storage address to be displayed or he can page forward and backward, one page at a time, using the PF keys. An offset value can be defined by the programmer so that each forward or backward paging request will reflect the data at the currently displayed storage address, adjusted by the offset value. This permits quick scanning through main storage—examination of a screen full of data every 1,000 or 5,000 bytes, for example, without having to view every byte in between.

MicroCICS includes a utility that sets breakpoints in programs under test. The breakpoint processor provides a menu screen that lists the options for setting traps within a program. When a trap is reached, program execution is interrupted and the debug processor takes control. The programmer can view and modify storage or alter the sequence of statement execution.

Positions for breakpoint options include when entering a module, when entering a command, when exiting a command, and on a nonzero (error) return code. On entering a command, the programmer may view and alter, if necessary, the parameters being passed to CICS. On exiting a command, the programmer can verify the CICS com-

mand return code and re-execute the command, if desired, by pressing a PF key. A breakpoint also may be set for any command that causes an abnormal (nonzero) return code.

The programmer may set 10 specific breakpoints by identifying the line number and module name of the desired program source code where the trap is to be set. Unlike the mainframe CICS debug tool EDF, MicroCICS can service breakpoints at any source statement within the transaction, not just at CICS commands. Periodic breakpoints may be set by specifying a cycle of the desired number of source statements for single step mode. After each cycle of statements has been executed, a breakpoint will be invoked. For example, the programmer may choose to stop after each statement is executed or after every tenth statement is executed.

The debug facility can be disabled for lower-level routines that have been linked into the CICS program via the CICS LINK command. The debug facility relinquishes control when the lower-level routine is called, and regains control automatically when the called routine returns control to the main program. This avoids retesting called routines that have been verified.

A useful feature of this facility allows the programmer to change the normal order in which statements are executed by specifying the line number of the desired statement. This permits rapid testing of routines that process multiple types of input. A routine could be run several times, with the programmer modifying the data to test a specific case each time before initiating the routine again. This would not be possible using mainframe CICS's EDF facility, which cannot run a single routine more than once unless the program under test were designed to do so. The user could test as many different data cases as desired before moving on.

This feature also allows the programer to bypass a section of code that is invalid. For example, a program may need to LINK to another program that is not presently available. If the LINK command is executed, the task will terminate with an error. A breakpoint can be set before executing the command; when the breakpoint has been reached, program execution can be resumed beginning with the statement after LINK to avoid generating the error.

An important advantage of being able to modify execution flow is that it can be used to circumvent a problem that requires recompiling to correct. If an incorrect statement causes program failure, the programmer can make data corrections and resume execution with the next viable statement.

MicroCICS defines PF keys to permit the viewing of the execute interface block (EIB) fields and the last output screen sent by the transaction. In addition, a PF key is assigned that will re-execute the last executed CICS command. If an error condition code was returned, the programmer often can modify the command parameters, retry the command, and continue testing.

SCREENS AND FILES

The screen generation utility within MicroCICS permits the programmer to draw the desired screen format interactively. Individual characteristics, such as attribute bytes, can be defined by the programmer for desired fields; MicroCICS will default the rest. Mapset and map information are created by MicroCICS using standard default assumptions, but they too can be viewed and modified by the programmer.

Once the screen has been described, MicroCICS can be used to generate CICS basic mapping support (BMS) macros. These macros then are used by MicroCICS to generate copy members (essentially, COBOL source include files) describing the map fields as well as to create the physical map format. By using BMS macros rather than creating its own mapping logic, MicroCICS is able to guarantee compatibility with the mainframe environment. Existing maps can be downloaded from the mainframe for modification or for use under MicroCICS.

MicroCICS contains a VSAM simulator that supports sequential (ESDS), keyed (KSDS), and relative record (RRDS) VSAM data sets. All VSAM functions available under CICS are supported under MicroCICS. These data sets can be updated, processed randomly or sequentially, and accessed through alternative keys via a PATH. Segmented records also are supported.

Files can be defined on-line using a file attribute maintenance menu that prompts the programmer for the necessary information. File information can be viewed, modified, or erased, and datasets may be renamed. Files also must be identified in MicroCICS's system tables, which parallel the mainframe CICS system table information.

Once the file is defined, the programmer keys in records interactively or loads a VSAM file from a sequential data file downloaded from the mainframe: existing mainframe files can be used to create MicroCICS data.

Conversely, MicroCICS VSAM files can be converted to sequential files which can be used for backup or uploaded to the mainframe system. IBM does not support the uploading or downloading of VSAM files through VMPCSERV or TSOSERV, hence the requirement that VSAM files first be converted to sequential files for transfer.

This menu-driven process allows records to be viewed or printed in whole or in part. New records can be entered interactively and existing records can be changed or deleted. With the file services provided by MicroCICS, test files can be created and loaded with data in a matter of minutes.

Limited support for DL/I is now available. It is limited to a user-exit system in which DL/I calls are intercepted by calls to a user-written routine that returns data to MicroCICS. Full DL/I support and support for IDMS databases in some form has been announced and will be available later in 1986.

HANDS-ON

Over and above the downloading of development work from a busy mainframe, MicroCICS is ideal for any situation that requires a secure, protected development environment. It can be used for prototype demonstrations without worrying about mainframe system availability or ill effects from other applications running concurrently. Single-user or low-volume applications can use MicroCICS for production work, shipping completed data to the mainframe via 3270 emulation. This could provide a micro-based on-line system for users whose applications might be too small to justify the cost of maintaining a mainframe CICS system.

A hands-on workshop for programmers in training is highly desirable although it requires a considerable commitment in time and resources. Companies must be able to guarantee a reasonable, consistent turnaround time and continuous test system availability.

MicroCICS can provide a dedicated workshop environment for classroom training that would eliminate waiting for mainframe turnaround time. Unlike the mainframe test system, availability and contention from other applications is never a problem under MicroCICS. One drawback is that if a CICS transaction dump is needed to teach programmers how to use such dumps in mainframe debugging, example dumps must be produced on the mainframe. (In ordinary MicroCICS program development, its debugging tools should make a dump listing unnecessary.)

Novice CICS coders often create storage violation problems that can be fatal to the mainframe CICS region and cause all testing to wait until the region can be restarted. Testing under MicroCICS isolates each task on a separate microcomputer and prevents this problem. Using MicroCICS for training protects the mainframe CICS region from such crashes while it provides uninterrupted workshop access.

MicroCICS can prove to be an extraordinarily useful tool in business. Its initial price, $4,495, may seem high from a microcomputer software perspective, but it is reasonable when compared to the cost of many mainframe software tools. The product is intended for the CICS developer who is able to see software cost in terms of its ability to do a job rather than a premium over the cost of the materials.

MicroCICS cannot produce CICS-formatted dumps, which may irritate longtime CICS programmers who use dumps as integral parts of their debugging processes. The CICS debug facilities make them unnecessary, however, and once programmers become familiar with MicroCICS, its lack of dumps should not hamper them.

As with any plug-compatible product in the IBM arena, the risk is ever-present that IBM will make some enhancement or change to mainframe CICS that will make its source code incompatible with MicroCICS. Unicorn Systems has done an admirable job of keeping up with IBM CICS enhancements thus far, but the lead time between IBM's enhancements and Unicorn's support of those enhancements within MicroCICS is unavoidable.

Overall, MicroCICS is a superior product that does what it says it will do, quickly and without error. It is one of few products now available that use the PC-370 power of the XT-370 and AT-370. Finally, it provides a good look at the long-promised "desktop mainframe" programming environment that will become common throughout corporate America in the future.

MicroCICS: $4,495
Unicorn Systems Company
3807 Wilshire Blvd.
Los Angeles, CA 90010
213/380-6974
CIRCLE 351 ON READER SERVICE CARD

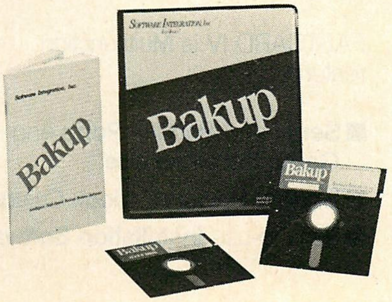
Brandy C. De Shazo has been designing command-level CICS systems since 1977. She has taught CICS classes nationwide for six years and is currently writing a self-study course on command level for her new Los Angeles-based company, Creative Online Systems.

PROTECT YOUR CORPORATE ASSETS.

Backing up your company's critical data is serious business. Consider the consequences if something unforeseen happened, and your valuable files were lost forever.

Software Integration's **BAKUP**™ software system stores your files in a safe place. Then, if disaster *does* strike, **BAKUP** allows you to retrieve everything exactly as it was.

BAKUP is a high-speed, state-of-the-art software utility designed to simplify, speed and manage the backup/ restore process on the hard disk of an IBM PC, XT, AT and compatibles.



BAKUP IS FAST.

The first backup of a 20MB disk takes less than 20 minutes -- then, daily backups take only a few minutes.

BAKUP IS EASY.

So easy, in fact, you might not even need **BAKUP**'s 200+ help screens or comprehensive users manual.

BAKUP is a trademark of Software Integration, Inc.; IBM PC, XT, AT are trademarks of IBM Corp.

BAKUP IS MORE.

No other backup utility offers these key features:

- **STORAGE MANAGEMENT** - **BAKUP** creates its own catalog to keep track of where your files are, when they were backed up, and how frequently. To restore, you simply browse through the catalog and mark the desired files; **BAKUP** tells you which diskettes to insert. **BAKUP**'s 50 pre-numbered labels make identification easy.

- **MEDIA INDEPENDENCE** - Because **BAKUP** supports most output devices, you only need one backup system.

SOFTWARE INTEGRATION, INC.
9800 S. Sepulveda Bl., #214
Los Angeles, CA 90045
213/776-3404

To Order Toll-Free: (800) 538-8157 Ext. 801 (USA)
(800) 672-3470 Ext. 801 (CA)

CIRCLE NO. 107 ON READER SERVICE CARD

BAKUP PROTECTION COSTS ONLY $179.95 . . . AND YOUR SATISFACTION IS GUARANTEED!! Just fill out the coupon below, or call our toll-free order number, and you can try **BAKUP** for 30 days on our money-back guarantee.

See us at COMDEX, Booth #C121-122

Yes, I want to protect my corporate assets with **BAKUP**.

Send me ___copies of **BAKUP**, with documentation and diskette labels, at $179.95 ea. backed by a 30-day, money-back guarantee. Add $5.00 shipping & handling. In California, add sales tax.

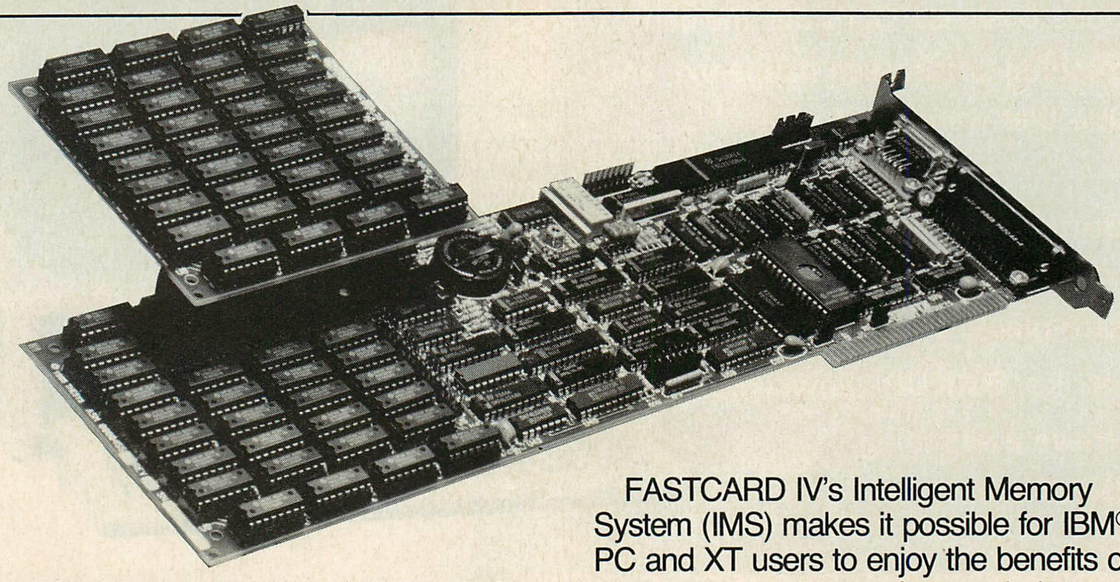
Name__________
Title __________
Company __________
Address __________
City __________
State/Zip __________
VISA/MasterCard # and Exp.:
____
Signature __________

☐ Check or money order enclosed for $ __________

TJ-12

FASTCARD IV

TWO MEGABYTES FOR YOUR PC–AND MORE!



The power of the Lotus® Expanded Memory Specification is now available in an intelligent, multifunction expansion board—THESYS' FASTCARD IV™.

FASTCARD IV's Intelligent Memory System (IMS) makes it possible for IBM® PC and XT users to enjoy the benefits of up to two megabytes of expanded memory with *all* the software in their libraries—*not just programs written especially for expanded memory.*

THESYS' IMS features include:

- **Disk Enhancement**—Frequently accessed disk information is stored in system RAM or expanded memory where it may be rapidly retrieved.
- **Printer Enhancement**—IMS allocates memory to serve as a holding area for information waiting to be printed.
- **Automatic Memory Allocation**—IMS monitors the computer's operations and allocates the system and expanded memory as needed for disk enhancement, printer enhancement, or other applications.

In addition, FASTCARD IV offers **Data Security** features vital to PC users:

- **Custom Password Security** enables the user to define a password that must be entered *before the computer will boot.*
- **Built-in Diagnostics** and **Auto Fault Tolerance** identify faulty memory and bypass it until it can be replaced.

FASTCARD IV is **Multifunction,** featuring:

- **Serial** and **Parallel Ports** and **Game Port Interface**
- **Battery-Powered Clock/Calender**
- Easy-to-use **Installation Software.**

THESYS
MEMORY PRODUCTS CORPORATION

For ordering information, contact:
THESYS Memory Products Corporation
7345 East Acoma Drive
Scottsdale, AZ 85260
(800)-327-8345
(602)-991-7356

CIRCLE NO. 186 ON READER SERVICE CARD

CHARLES C. EDWARDS

INLINE Interrupts

Turbo Pascal's interrupt handling feature requires some minor adjustments before it will execute properly.

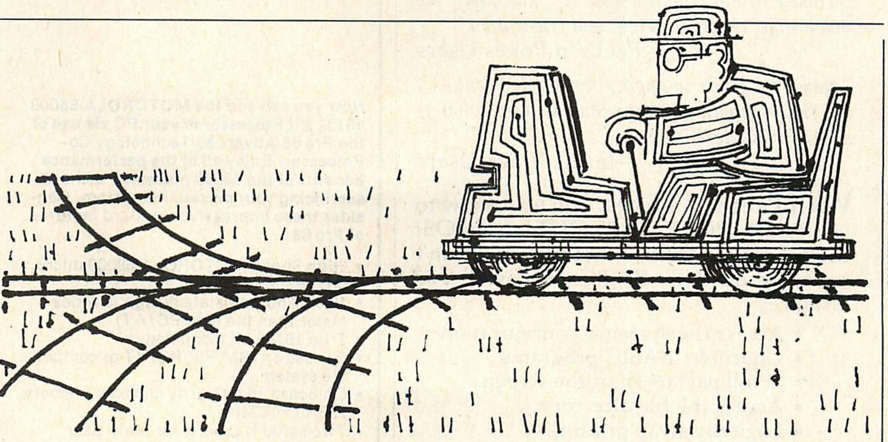


ILLUSTRATION • MACIEK ALBRECHT

A most interesting feature of Turbo Pascal is its ability to generate interrupt handling code. However, the product's documentation of this feature is incorrect in at least one particular. Offered here are guidelines to writing Turbo interrupt handlers, plus clarification of the information that appears in the *Turbo Pascal Reference Manual* (section B.1.11 in version 2.0 and on page 214 in version 3.0). The example used in this article is a dumb terminal emulator that employs interrupts to buffer input from the PC's serial port. This program is based on the assembly language program written by Chris Dunford for "Interrupts and the IBM PC" (*PC Tech Journal*, November/December 1983, p. 173 and January 1984, p. 144), a two-part article that offers a complete discussion on interrupt handling.

The first step in developing such a routine is to read the information on interrupt handling in the Turbo Pascal documentation. The 2.0 manual explains that the interrupt procedure must have no parameters since no way exists to specify the parameters when the procedure is called. The 3.0 manual does not repeat this statement, but the situation remains true nonetheless. The procedure simply will be invoked from whatever code the PC happens to be running when the interrupt occurs. This also makes it necessary to save all the hardware registers upon entry and to restore them when the procedure is finished. Borland International, Turbo Pascal's creator, suggests using the lines of code that are shown in figure 1 to accomplish this task.

Two problems occur with the code in figure 1. The first, and most serious, problem is that the procedure will not return to the interrupted code due to extra instructions that Turbo Pascal places at the beginning of each procedure. The second difficulty is that the interrupt procedure does not have access to the data in the program's outer block. A closer examination of these problems reveals the steps to be taken toward their solution.

Turbo Pascal places the following three extra instructions at the start of each procedure:

```
PUSH BP
MOV BP,SP
PUSH SP
```

(The reason for this is explained in section B.1.12 of the 2.0 manual and on page 216 of the 3.0 manual). The manuals say that all variables local to a procedure reside in the stack segment and their offsets are relative to the BP register. The effect of this code is to save the old value of the BP register and to initialize BP to point to the current stack offset. Thus, it is possible to address all the local variables via BP.

This, however, raises a problem in an interrupt handling procedure. Such a procedure cannot terminate with a simple RET (RETurn) instruction because the interrupt causes the values of the flags to be PUSHed on the stack along with the return address. An interrupt procedure must be terminated with an IRET (Interrupt RETurn) instruction, which causes the flags to be POPed from the stack.

Turbo PUSHed the BP and SP registers onto the stack prior to entering the procedure; therefore, they must be POPed off before the IRET can be executed safely. This is done by placing the following instructions before the IRET:

```
MOV SP,BP
POP BP
```

The second problem is more subtle and will not occur in all cases. A programmer who is writing an interrupt procedure in Turbo Pascal probably will want to pass data between the interrupt procedure and the main program. For example, in the terminal emulator program that accompanies this article, the data received from the serial port must be given to the main program to be processed.

Refer again to section B.1.12 of the 2.0 manual; it says that global data reside in the data segment and that its offset is relative to the DS register. In other words, in order to access global data from the interrupt procedure, the DS register must be properly initialized. Because an interrupt procedure can be invoked from anywhere, the value of the DS register upon entry to the procedure is unknown; therefore, it must be initialized.

This process is facilitated by a misnamed feature of Turbo Pascal called *typed constants*, which are not constants at all. Rather, they are variables that are

assigned an initial value. The following is an example of a typed constant:

```
Const Thirteen:Integer = 13;
```

This statement declares a typed constant called Thirteen with an initial value of 13. Variables, which have no initial value, differ from typed constants in another respect. While variables reside in the data segment, typed constants reside in the code segment where they can be accessed without knowledge of the data segment address.

While data could be passed back to the main program by storing it in a typed constant, a more flexible approach is available. If the program's outer block stores the value of the DS register in a typed constant, the interrupt procedure can use this to initialize the DS register. The interrupt procedure now is able to address the entire data segment of the main program. The outer block can determine the value of the DS register by using the Turbo construct DSEG. This method is illustrated in the following program fragment:

```
Program Interrupt_Driven;
  Const Data_Segment:Integer = 0;
  {Other declarations go here}
Begin {Outer Block}
  Data_Segment: = DSEG;
  {Other statements go here}
End.
```

In addition, it is necessary to modify the two INLINE statements in figure 1 to save and restore the old value of the DS register as well as to load the value of the Turbo data segment. The resulting INLINE statements are shown in figure 2.

Now all the tools necessary to building an interrupt driven data communications program are assembled. Listing 1 is a fully functional dumb terminal program. It simply sends anything it receives from the keyboard to the serial port, and anything it reads from the serial port is sent to the screen; this program can be tailored to individual specifications. The general method of handling interrupts in Turbo Pascal can be applied to any kind of interrupt, not simply hardware interrupts from a serial port. The saddest lesson in all of this is that the user cannot always trust what he reads in a product manual.

Charles C. Edwards is a private mainframe consultant to a large international organization. He has 10 years of experience with Burroughs mainframes and 5 years with microcomputers. He earned a bachelor of science in mathematics from the Stevens Institute of Technology in Hoboken, New Jersey.

FIGURE 1: *Borland's Entry and Exit Code*

```
          Entry                              Exit

Inline ($50/     {PUSH AX}      Inline ($07/     {POP ES}
        $53/     {PUSH BX}              $5E/     {POP SI}
        $51/     {PUSH CX}              $5F/     {POP DI}
        $52/     {PUSH DX}              $5A/     {POP DX}
        $57/     {PUSH DI}              $59/     {POP CX}
        $56/     {PUSH SI}              $5B/     {POP BX}
        $06/     {PUSH ES}              $58/     {POP AX}
        $FB);    {STI}                  $CF);    {IRET}
```

This code will fail because it does not remove BP and SP from the stack on procedure exit. In addition, the value of DS is unknown; thus it cannot be used to access global data.

FIGURE 2: *Revised Entry and Exit Code*

```
                    ENTRY
Inline ($50/                      {PUSH AX}
        $53/                      {PUSH BX}
        $51/                      {PUSH CX}
        $52/                      {PUSH DX}
        $57/                      {PUSH DI}
        $56/                      {PUSH SI}
        $06/                      {PUSH ES}
        $1E/                      {PUSH DS}
        $FB/                      {STI}
        $2E/$A1/Data_Segment/     {MOV AX,
                                  [CS:Data_Segment]}
        $8E/$D8);                 {MOV DS,AX}

                EXIT
Inline ($1F/          {POP DS}
        $07/          {POP ES}
        $5E/          {POP SI}
        $5F/          {POP DI}
        $5A/          {POP DX}
        $59/          {POP CX}
        $5B/          {POP BX}
        $58/          {POP AX}
        $8B/$E5/      {MOV SP,BP}
        $5D/          {POP BP}
        $CF);         {IRET}
```

Here, BP and SP are removed from the stack on exit. The entry code initializes DS for global data access by moving the value at CS:Data_Segment into DS.

LISTING 1: COMM.PAS

```
Program Comm;

Type INS8250 = Record            {Define serial port registers}
               THR:Integer;       {Transmit holding register}
               RBR:Integer;       {Receive holding register}
               IER:Integer;       {Interrupt enable register}
               LCR:Integer;       {Line control register}
               MCR:Integer;       {Modem control register}
               LSR:Integer;       {Line status register}
               MSR:Integer;       {Modem status register}
             End;

     Registers = Record    {Define the 8086 chip registers}
                 AX,BX,CX,DX,BP,SI,DI,DS,ES,Flags:Integer;
                 End;

     Baud = (B110,B150,B300,B600,B1200,B2400,B4800,B9600);

     Parity = (None,Odd,Nevermind,Even);

     {Define addresses for COM1 & COM2}
Const RS232:Array [1..2] of INS8250 =
          ((THR:$3F8;RBR:$3F8;IER:$3F9;
            LCR:$3FB;MCR:$3FC;LSR:$3FD;MSR:$3FE),
```

```
          (THR:$2F8;RBR:$2F8;IER:$2F9;
           LCR:$2FB;MCR:$2FC;LSR:$2FD;MSR:$2FE));

Var COM:Byte;                         {COM1 = 1 and COM2 = 2}
    RS_Error:Byte;                    {Error from serial port}
    Buffer:Array [0..4095] of Byte;   {4K data comm. buffer}
    Buf_Head,Buf_Tail:Integer;        {Head/tail buffer pointers}
    Regs:Registers;                   {Passes parms to MSDOS}
    Char_In:Integer;                  {Char read from serial port}

Const Data_Segment:Integer = 0;       {Holds contents of DS reg.}


Function Keyin:Integer;
Var Key:Byte;

    {This function reads in one keystroke from the keyboard.
     If a normal key was pressed, the value is returned in the
     low order byte of the integer value. If an "extended" key
     (such as a function key) was pressed, the low order byte
     contains a zero and the high order byte contains the
     extended code}

  Begin
    Regs.AX:=$0700;
    MSDos(Regs);
    Key:=Lo(Regs.AX);
    If Key = 0 then
      Keyin:=Keyin*256
    else
      Keyin:=Key;
  End;  {of Keyin}


Procedure RS232_Interrupt;

    {This procedure handles interrupts from the serial port.
     RS_Error is set to reflect any error result from the port,
     but nothing is done with this. The character is read in
     from the port and stored in the buffer and Buf_Tail is
     incremented.  No test is made for buffer overflow.
     After all registers have been pushed, the data segment
     containing global data is loaded into DX from the
     typed constant Data_Segment.}

  Begin
    Inline ($50/                      {PUSH AX}
            $53/                      {PUSH BX}
            $51/                      {PUSH CX}
            $52/                      {PUSH DX}
            $57/                      {PUSH DI}
            $56/                      {PUSH SI}
            $06/                      {PUSH ES}
            $1E/                      {PUSH DS}
            $FB/                      {STI}
            $2E/$A1/Data_Segment/     {MOV AX,[CS:Data_Segment]}
            $8E/$D8);                 {MOV DS,AX}
    RS_Error:=Port[RS232[COM].LSR] and $1E;
    Buffer[Buf_Tail]:=Port[RS232[COM].RBR];
    Buf_Tail:=(Buf_Tail+1) mod 4096;
    Inline ($FA);                     {CLI}
    Port[$20]:=$20;   {Clear Interrupt flag}
    Inline ($1F/                      {POP DS}
            $07/                      {POP ES}
            $5E/                      {POP SI}
            $5F/                      {POP DI}
            $5A/                      {POP DX}
            $59/                      {POP CX}
            $5B/                      {POP BX}
            $58/                      {POP AX}
            $8B/$E5/                  {MOV SP,BP}
            $5D/                      {POP BP}
            $CF); {Return}            {IRET}
  End;  {of RS232_Interrupt}


Procedure RS232_Init(Speed:Baud;P:Parity;Stop,Length:Byte);

    {This procedure uses Interrupt 14
     to initialize the serial port.}

  Begin
    Regs.DX:=Com-1;
    Regs.AX:=Ord(Speed)*32 + Ord(P)*8 + (Stop-1)*4 + Length-5;
    Intr($14,Regs);
  End;  {of RS232_Init}


Function RS232_Out(Param:Byte):Boolean;
Var Counter:Real;

    {This function outputs a byte to the serial port.
     In the event of a timeout, a value of TRUE is returned.
     A timeout occurs if we cannot get Data Set Ready,
     Clear to Send, and an empty transmit holding register
     after 65535 attempts.}

  Begin
    Counter:=0;
    Port[RS232[COM].MCR]:=$0B;
    While
      ((Port[RS232[COM].MSR] and $30) <> $30)  {DSR and CTS}
    and
      ((Port[RS232[COM].LSR] and $20) <> $20)  {Tr. Reg empty}
    and
      (Counter < 65535.0) do                   {Timeout}
        Counter:=Counter+1;
    If Counter = 65535.0 then
      RS232_Out:=True         {We timed out}
    Else
      Begin
        Port[RS232[COM].THR]:=Param;
        RS232_Out:=False;
      End;
  End;
```

Andy says I'm nuts to send you Q-PRO 4 for $50

Multi-User 4th generation data base language can help any professional developer multiply income

To prove it, I'll send you the complete Q-PRO 4 applications development system, the full Q-PRO 4 manual, and even refund the token $50 security deposit if you don't agree

Hard to believe, but true.

Now you can save hundreds of programming hours and complete more jobs profitably.

As part of an international promotion campaign, we are distributing 5000 copies of the Q-PRO 4 data base language for FREE trial.

We will rush Q-PRO 4 to the first 5000 professional developers who send us a token $50 security deposit (100% refundable).

Over 10,000 professionals now save millions of programming hours with Q-PRO 4.

Multi-user Q-PRO 4 is *the* most widely used *professional* data base language with no restrictions. It has already earned rave reviews from developers in the USA, Japan, Canada, Great Britain, Australia, and 34 other countries.

Multiply your productivity with Q-PRO 4

The screen generator, file management system, and super efficient syntax have been optimized for the professional developer.

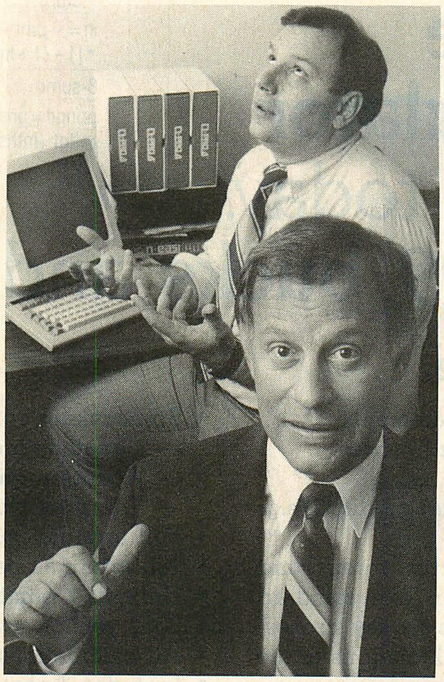
Complete Q-PRO 4 system for trial use . . . not a demonstrator

You will get the complete Q-PRO 4 system with all the features that pros now use worldwide. One exception . . . the run-time module has been altered to make it suitable for trial use.

89% of the professionals who try Q-PRO 4 . . . buy Q-PRO 4

Professional developers around the world tell us *Q-PRO 4 has no true competition.* So, instead of spending a million dollars on Q-PRO 4 advertising claims, we're sending you Q-PRO 4.

You try Q-PRO 4. You agree that Q-PRO 4 will save you hundreds of programming hours . . . or send it back for return of your $50 security deposit. No questions asked!



Bob Roth, Q-N-E International's president and Andy Erlanger, author of Q-PRO 4, at computer

Fabulous Q-PRO 4 Offer and Guarantee

- You get a complete FREE trial copy of Q-PRO 4.
- You get a complete manual.

Your obligation

- You send a security deposit of $50 plus $5 handling (credit cards accepted).
- You try Q-PRO 4.

Our 100% Guarantee

- We guarantee your complete satisfaction with Q-PRO 4.
- If for any reason, you return to us the Q-PRO 4 system and Q-PRO 4 manual within 30 days of receipt, we will return your full $50 security deposit.

Bob and Andy

CIRCLE NO. 173 ON READER SERVICE CARD

Only the Q-PRO 4 data base language gives professional developers these twelve blockbusters to multiply your productivity and income.

DATA BASE
1. FILE TYPES: Random, Sequential, Index Sequential, Relational
2. NUMBER OF OPEN FILES: 255
3. RECORD SIZE: Unlimited
4. NUMBER OF FIELDS: Unlimited

LOCAL AREA NETWORKS
5. FULL FILE AND RECORD LOCK

PORTABILITY
6. 8-BIT TO 16-BIT TO 8-BIT

FORMATTED DATA ENTRY
7. FULL WITH FIELD ATTRIBUTES

REPORT GENERATOR
8. FULL . . . CREATED ON SCREEN

MEMORY VARIABLES
9. UNLIMITED

PROGRAMMABLE FUNCTION KEYS
10. TWENTY-ONE

QUERY LANGUAGE
11. COMPANION PACKAGE – $250

COMPILER
12. UNDER DEVELOPMENT

Q-PRO 4 PURCHASE PRICES
FULL PRODUCTION SYSTEM:
Single-user – $595
LAN/Multi-user – $795
Author's lock up package available

Be one of the lucky 5000 pros that takes advantage of this FREE trial offer. You risk nothing. Call now with credit card number.

For the little more daring . . . order the full production system now . . . it comes with 100% money-back guarantee, too.

Questions? Call us! Ask about surprise bonus offer.

136 Granite Hill Court, Langhorne, PA 19047
(215) 968-5966 Telex 291-765

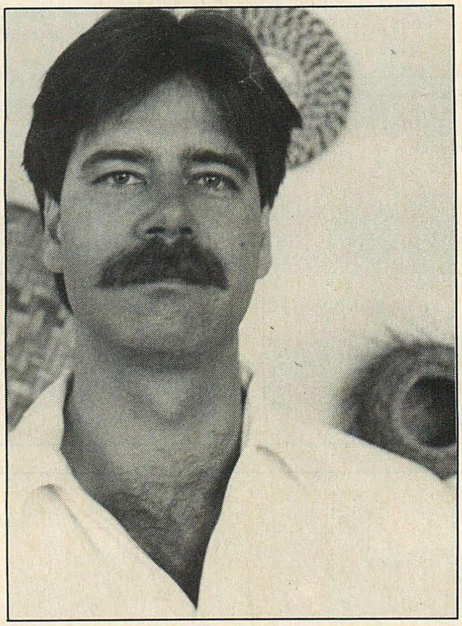
q-n-e international™

```
Procedure Initialize;

    {This procedure does the following:
     1)  Changes the appropriate interrupt vector to point to
         RS232_Interrupt.
     2)  Enables the appropriate interrupt on the PC interrupt
         controller.
     3)  Enables the "data ready" interrupt on serial board.}

    Begin
      Regs.AX:=$250D-Com;
      Regs.DS:=CSEG;
      Regs.DX:=Ofs(RS232_Interrupt);
      MSDos(Regs);      {Set up interrupt vector}
      If COM = 1 then
        Port[$21]:=Port[$21] and $EF  {Enable IRQ4}
      else
        Port[$21]:=Port[$21] and $F7; {Enable IRQ3}
      Port[RS232[COM].LCR]:=Port[RS232[COM].LCR] and $7F;
      Port[RS232[COM].IER]:=1;
      Port[RS232[COM].MCR]:=$08;
    End;  {of Initialize}

Procedure Cleanup;

    {This procedure disables the "data ready" interrupt.}

    Begin
      Port[$21]:=Port[$21] or $18;
      Port[RS232[COM].LCR]:=Port[RS232[COM].LCR] and $7F;
      Port[RS232[COM].IER]:=0;
      Port[RS232[COM].MCR]:=0;
    End;  {of Cleanup}

Procedure TTY;

    {This procedure provides basic terminal emulation.
     Characters read from the keyboard are sent to the serial
     port.  Characters read from the serial port are sent to
     the screen.  The procedure is terminated by typing a
     CTRL-Z.}

    Begin
      ClrScr;
      Repeat  {This code does NOT suppress NULs!}
        While Buf_Tail <> Buf_Head Do
          Begin   {There is data in the buffer}
            Write(TRM,Char(Buffer[Buf_Head]));
            Buf_Head:=(Buf_Head+1) mod 4096;
          End;
        If Keypressed then
          Begin   {Process a keystroke}
            Char_In:=Keyin;
            If Char_In <> 26 then  {Ctrl Z to end}
              Begin
                If Char_in = 13 then
                  Write(TRM,Char(10));
                If RS232_Out(Lo(Char_In)) then
                  Writeln(TRM, '*** RS232 Timeout ***');
              End;
          End;
      Until Char_In = 26;  {Ctrl Z}
    End;  {of TTY}

Begin  {Outer Block}
  COM:=1;   {Default to COM1}
  Buf_Head:=0;
  Buf_Tail:=0;
  Data_Segment:=DSEG;   {Store the contents of the DS register}
  {Init port to 1200 Baud, Even parity, 1 Stop bit, 7 Data bits}
  RS232_Init(B1200,Even,1,7);
  Initialize;
  TTY;
  Cleanup;
End.
```

"I Program In BetterBASIC And I Recommend It."

Neurometrics, Inc. Lifescan (tm) brainwave monitor required an optional module that would allow anesthesiologists to perform enhanced data analysis using PCs. Because source code is provided for program modification, BASIC was selected due to its widespread use. But BASIC lacked strong file management and modularity. That's when I heard about BetterBASIC. BetterBASIC gave me the file management and modular features needed to break my large programs into logical units. Now I program in BetterBASIC and I recommend it.

David Geving, Systems Analyst - Neurometrics Inc. - San Diego, CA



640K Now you can use the full memory of your PC to develop large programs.
STRUCTURED Create well organized programs using procedures and functions that are easily identified and understood and completely reusable in future programs.
MODULAR Use procedures and functions grouped together to form "library modules" which are then available to you or anyone else for future use.
EXTENSIBLE Create your own BetterBASIC modules which contain BetterBASIC extensions. This feature coupled with the easy-to-use Assembly Language support, makes this an ideal OEM language.
INTERACTIVE BetterBASIC acts like an interpreter because it responds to the users' commands in an immediate mode. However each statement is actually compiled as it is entered.
COMPILED Each line of the program is compiled as it is entered into the computer's memory rather than interpreted at runtime.
RUNTIME SYSTEM The optional Runtime System generates stand alone EXE. files allowing for the distribution of products written in BetterBASIC with no royalties.
SUPPORTS Windows, Graphics, DOS and BIOS ROM calls, Chaining, Overlays, Local and Global Variables, Recursion . . . and more.
BetterBASIC Runs on IBM PC, XT, AT and all IBM-compatibles. Ask your local dealer for BetterBASIC or call 1-800-225-5800 in Canada call 416-469-5244. Also available for the Tandy 1000, 1200, AND 2000 AT Tandy/Radio Shack stores.

Summit Software Technology, Inc.™
P.O. Box 99, Babson Park
Wellesley, MA 02157

See Us At COMDEX Lifeboat's Booth #1392

PRICES:

BetterBASIC	$199
8087/80287 Math Module	$99
BTrieve™ Interface	$99
Runtime System	$250
Sample Disk with Tutorial	$10

Better BASIC.

Because It's The Best.

CIRCLE NO. 195 ON READER SERVICE CARD

MasterCard, Visa, Checks, Money Order, C.O.D. accepted and P.O. on approval.

(If you're using BetterBASIC and would like to be featured in one of our ads, please write to the Director of Advertising at Summit.)

Proprietary Rights and Software

The ABA has released a report on proprietary rights in software. The conclusion: the issues are so complex a consensus cannot be reached.

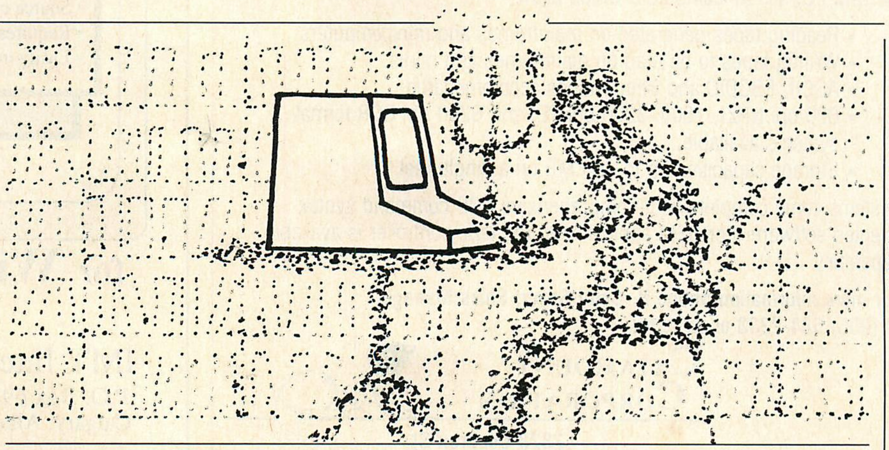


ILLUSTRATION • MACIEK ALBRECHT

Two years ago the American Bar Association's Computer Law Division of the Section on Science and Technology was already at work resolving some of the uncertain legal issues surrounding the protection of software. It had begun a study of proprietary rights in software with a view to recommending legislation through the Bar Association (see "The Basic Tools of U.S. Intellectual Property Law," Legal Brief, Max Stul Oppenheimer, November/December 1983, p. 213). The first step in that process recently has been concluded with the publication of a report, detailing the results of a poll of the committee membership on selected issues and the committee chairman's own analysis of the results and his recommendations.

The report "does not necessarily represent any consensus of opinion of members of the committee, of the section . . ., of the ABA or of any other organization." (I, for one, would like to disassociate myself from the statement in the report that "when a computer program is input from a floppy disk or other external storage into a computer for operation a number of separate copies of the program are created in the various components of the computer's memory.") However, the ABA report does present what the committee considers to be the major issues, from a legal point of view, that are currently confronting the software industry.

Issue 1: The first sale doctrine. Section 109 of the Copyright Act permits the owner of a copyrighted work to dispose of it as he chooses. The purchaser of a copy of a book can decide to read, lend, sell, rent, or throw it away. The committee was asked to consider whether Section 109 should be made inapplicable to software (a result that most license agreements attempt to achieve contractually). The membership split evenly on the issue. The committee chairman recommended that the copyright law be amended specifically and directly to prohibit rental of software, which he perceived to be the basic problem created by the application of the first sale doctrine to software.

Issue 2: The meaning of copying a computer program. The proposal was made that the copyright law be amended to provide that loading a computer program into RAM constituted copying the program. This issue is significant because if loading a program into memory is deemed to create a *copy*, then the general copyright law prohibition of copying must be specially modified with respect to computer software. Otherwise, loading a program from disk into memory would be unlawful.

The exact wording of such a special modification can give rise to other problems. One example presented by the report is whether a distinction should be made between local area networks and multiuser computers. The report states that under the proposed redefinition, multiuser computers would not violate copyright, but LANs would. However, a technical reading of the proposed redefinition seems to leave a problem for even a single-user system because it also creates a copy of the program in RAM.

Current Section 117 of the Copyright Act authorizes the creation of a copy (or adaptation) of a computer program as "an essential step in the utilization of the computer program. . . ." If Section 117 is not amended, it would arguably continue to protect the LAN application even under the proposed redefinition of *copy*. The report's interpretation is that most people believe the law already provides that loading a program into RAM creates a copy.

Issue 3: Reverse engineering. At the crux of this issue is the question of whether the Copyright Act should be amended to prohibit disassembly, decompilation, printing, or other reverse engineering of software. Again, most license agreements attempt to impose these restrictions. As the report explains, however, "Section 117 can be interpreted to provide that there is a type of 'fair use' for software buyers such that they can take steps to understand the inner workings of software in order to make adaptations. The threat of this occurring would jeopardize trade secret protection. Therefore, many software companies go through complex license arrangements . . . Many of these arrangements are through self-executing licenses of questionable enforceability." The ABA committee opposed the ban on reverse engineering.

Issue 4: Secure deposit of software. Registration of a copyright requires the deposit of a copy of the work. Unless special

provisions are made, the deposit copy is available for public inspection. In the case of software, Copyright Office regulations permit deposit of less than the complete work, and prudent software authors often are advised to structure their programs so they permit technical compliance with the regulations while submitting only nonessential portions of the program. The constitutional basis for the Copyright Act, Article 1, Section 8, is intended to "promote the progress of science and the useful arts." It is debatable whether such promotion is accomplished if the Copyright Act simply encourages authors to create or whether the constitutional purpose requires disclosure of the underlying creative ideas so that other persons may build on them.

The proposal before the ABA committee would have required filing of the complete source code plus annotations and documentation, which would have become completely public after 5 years unless the owner took steps to renew protection for an additional 15 years. An offshoot of the proposal would have allowed this filing to serve as a national source-code escrow system to attempt to deal with the as-yet-unresolved question of whether a license agreement survives the bankruptcy of the licensor. The committee split evenly, with some opponents of the proposal apparently feeling that copyright protection should require immediate public filing and others feeling that 20 years of secrecy did not afford sufficient protection to the software authors.

Issue 5: A new copyright symbol. Although hardly the intellectual peer of the other issues, this one is of practical consequence. For some reason, computer manufacturers have not included a key that produces the standard c-in-a-circle copyright symbol. Typing the word *copyright* is simple enough, but compliance with some laws of international protection requires the symbol instead of the word. The committee overwhelmingly supported using the symbol *c* in place of the standard circle. That did not solve the problem of international protection, but it did give the committee an issue upon which 70 percent of them could agree.

Issue 6: Coexistence of copyright and trade secrecy. In the opinion of some lawyers, trade secrecy and copyright protection are inconsistent. Until 1976, federal copyright protection required publication of the material to be protected. The 1976 copyright revisions did away with the publication requirement and, at the same time, preempted state laws dealing with similar protections as the federal copyright law. Clearly, common law copyright (which, before 1976, protected unpublished works) was gone, but it did not really matter because federal copyright took its place.

The argument has been put forth, however, that state trade secret laws are also preempted by the copyright law because they attempt to protect the same rights and because trade secrets theoretically can extend beyond the statutory copyright period. This issue is of particular practical consequence because many software publishers distribute object code and attempt to protect the source code as a trade secret (while claiming copyright protection on both). The committee supported clarifying that the Copyright Act does not preempt state trade secret laws.

As I predicted (with no great risk of error) two years ago in the "Basic Tools of U.S. Intellectual Property Law" article mentioned above, the process of dealing with these software issues is moving ahead slowly. What emerges from the ABA committee report is that none of the important policy issues garners a strong consensus. A skeptic might suggest that this is because each side to the controversy retains lawyers, so an even split within a committee of lawyers is hardly surprising.

A more charitable view is that the policy issues are difficult. Courts have repeatedly held that the Copyright Act protects expression only. Some have argued that the copyright monopoly is granted on expression so that ideas will become public. Software is unlike any other form of copyrightable expression in that identical results can be achieved using dissimilar code. To note that copyright law is rife with ambiguities when confronted with this novel situation is only the beginning of the analysis. Even if it is agreed that these ambiguities should be resolved legislatively, the problem of finding the appropriate resolution remains difficult. The ABA report reveals that its committee has yet to decide on whether to support legislative reform of the copyright system, because the members cannot agree on the goals of such reforms.

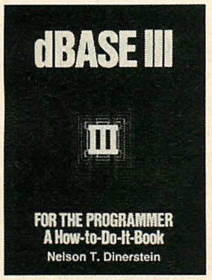
Have courage. Even a journey of 10,000 miles begins with the appointment of a subcommittee and the publication of a report.

Max Stul Oppenheimer, PC, is a partner in the law firm of Venable, Baetjer & Howard, located in Baltimore, Maryland.

An Update on dBASE

Programming in dBASE III and conversions from dBASE II to III are discussed fully in this new work.

dBASE III for the Programmer—A How-To-Do-It Book
Nelson T. Dinerstein (Scott, Foresman and Company, Glenview, IL, 1985) 312 pages; softcover, $19.95



An expansion of the author's previous work (*dBASE II for the Programmer*), this book addresses the programming aspects of dBASE III, Ashton-Tate's IBM PC version of its best-selling database management program. dBASE III incorporates several major changes to the old favorite, many of them involving its built-in programming language. The book does assume some programming experience on the part of the reader, although this experience does not necessarily have to be in the dBASE II or dBASE III languages. The experience requirement is relative, however; most "home" BASIC programmers will profit from the material.

The early portion of this book is dedicated to what the author calls a dBASE III primer, a short course in dBASE III commands, functions, and full-screen editing. The syntax for each command, along with a short description of its operation, is provided. Many of the commands are demonstrated in examples that are simple but effective. One helpful feature of this chapter is a two-page table that gives the reader a brief description of each command.

Occasionally Dr. Dinerstein includes a warning about the operation of some command that could cause problems for the programmer. Most of these are quite handy, but some are overly cryptic, even erroneous. For example, one of the warnings says, "The MODIFY COMMAND does not appear to work correctly all the time." This prompts obvious questions: When does it fail? What does it do when it fails? Does it erase the command file or the hard disk? If the author knows the answers to these questions, he chose not to pass them on to the reader. In another instance, he warns, "In order to use the RUN statement, you will need COMMANDS.CMD (sic) on your disk . . ."

The majority of the book deals directly with programming under dBASE III. Although the emphasis is on working with database files, the book offers general tips that can help in writing more elegant and efficient programs. The author gives examples of the coding style that he prefers, but explains other methods of performing many of the same functions. An entire chapter is devoted to design principles with the author demonstrating how an assemblage of data for solving a particular problem is divided into several database files and how the relationships between fields are determined. He gives examples of when to include the same field in two different files in order to speed up processing even though it makes updating the files more difficult.

One chapter illustrates a sample problem involving the performance of a group of salesmen and the dBASE III programs written to help them. Two database files are built, one containing information on the salesmen and the other with data on their accounts. Programs then are developed to enter, modify, and delete the records in these files. Throughout the development, the author explains how and why each part of the program was included.

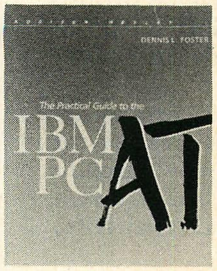
Many users of dBASE II intend to convert their programs to dBASE III and are concerned with the difficulties they may encounter. Dr. Dinerstein discusses the conversion of programs and database files using both the dCONVERT program included with dBASE III and the manual method. He describes the advantages and disadvantages of each but does not make a recommendation.

Reading about how to program can provide basic information about a language, but it does not teach the user how to program. The only way to learn programming is to write programs. The best that can be hoped for from a book on the subject is that it will include many good examples. *dBASE III for the Programmer* does just that, in particular with an 80-page appendix entitled "A Large System In dBASE III" consisting of sample programs from an account payables system. Documentation explains the functions used. The other two appendixes list the technical specifications of dBASE III and error messages with probable causes and solutions.

dBASE III for the Programmer is a good choice for the user who wants to increase his programming flexibility by adding the dBASE III language to his repertoire, who wants to learn the conversion from dBASE II to dBASE III, or who simply seeks an understanding of how programs are devised to solve database management problems. The author discusses each command and delves into program design—an area most books of this type ignore completely.

—BRUCE T. FILBECK

The Practical Guide to the IBM PC AT
Dennis L. Foster (Addison-Wesley Publishing Company, Inc., Reading, MA 1985) 455 pages; softcover, $19.95



This book may be a solution looking for a problem: it seems to be a beginner's guide to the PC/AT, but very few first-time computer users will start with an AT. The author seems also to have had problems defining his intended audience because he neglected to include a preface or an

introduction paragraph or even a "This book is intended for . . ." sentence. Reading the book did not help to clear up any of the confusion.

Foster begins the book with a description of the hardware that is generally accurate, although simplistic. In addition, some of it is misleading or incorrect. In the section on data storage, he states, "IBM hard disks utilize a technology in which a rigid platter is fixed inside a *cylinder* . . ." Later, in a section entitled "Partitioning the Hard Disk," he explains, "If you visualize the disk as a phonograph record, each circular groove is a single track. Each song is represented by a separate grouping of tracks. Imagine that all the songs are of identical length, and you'll have a pretty fair idea of how the tracks on a hard disk are organized into *cylinders*." For the user who correctly thought that cylinders were so called because a specific track on each of several disk surfaces (when stacked one above the other) resembled a cylinder, Foster's two differing (and incorrect) representations cause confusion. This confusion can easily undercut the book's credibility for the skeptical reader.

In another section, the author offers his explanation of why a hard disk frequently is called a *Winchester*. He states, "Because the first working model utilized two platters of 20 megabytes each (i.e., 20-20), IBM engineers dubbed the new drive the 'Winchester' disk . . ." This may be true, but either Foster or the IBM engineers apparently were not aware that the Winchester Arms Company manufactured many .30-.30s, but very few .20-.20s.

The book discusses TopView, DOS 3.0/3.1, editing with DOS (highlights about EDLIN), XENIX, and BASIC 3.0, all as they relate to the AT. It has seven appendixes, including a list of ASCII codes, glossaries of DOS, XENIX, and BASIC commands, and, finally, one called "Hardware Troubleshooting."

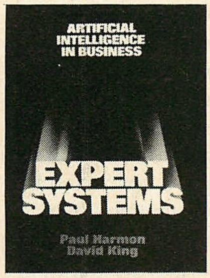
This last appendix lists nine possible problems and remedial actions. The author frequently suggests that the user "look for bent or broken pins—especially pins that may be bent beneath a chip." Pulling out chips to look for pins bent beneath them is a task that is better left to qualified technicians, yet Foster mentions consulting a technician only as the last resort for four of the problems. A user who is capable of troubleshooting hardware problems on his AT does not need this appendix. A user who does not have this capacity will find that this appendix is not much

help, and that following its advice certainly could make matters worse.

The arrangement of information within each chapter and section is haphazard and disorganized. Foster disregards transition as he jumps from one topic to another; he even repeats several paragraphs verbatim under different headings. With its poor organization, erroneous statements, and generally low information content, *The Practical Guide to the IBM PC/AT* seems to attempt to take advantage of the current boom in personal computer books. Many *good* computer books make it to the shelves at the local bookstore. This is not one of them.

—BRUCE T. FILBECK

Expert Systems: Artificial Intelligence in Business
Paul Harmon and David King (John Wiley and Sons, Inc., New York, NY, 1985) 280 pages; softcover, $16.95



If first impressions are deceiving, the title of this book might lead the wary reader to believe that it is not a serious work on computer science, what with "business" tacked on the end. This is not the case, however. Harmon and King have written a fine introductory text on knowledge-based or *expert* systems, condensing file cabinets of raw material into a 280-page gem. It is a comprehensive work; the authors touch on most of what has been or can be said about expert system design and application.

Expert Systems is intended for the business community, and it is equally appropriate for management and non-management users. The introduction discusses the motives for expert system development and is almost starry-eyed in its estimation of the potential benefits offered by AI engineering. It gives a brief historical accounting of computer system development that includes a chart of significant AI events, and offers simple comparisons of common computer subjects and expert concepts. In an abbreviated discussion on the market for expert systems, the authors use charts to show where they think the technology is headed. Also, the differences between generic large and small AI systems are distinguished.

In the first section, Harmon and King use the ever-popular MYCIN suc-

cess story to exemplify expert system development and data structure fundamentals used in knowledge-based systems. MYCIN is a software system designed to "aid physicians in the diagnosis and treatment of meningitis and bacteremia infections." A sample consultation shows how an expert system would interact with a user. An analysis on the success of MYCIN is included.

The next section deals with human problem solving, focusing on the way in which humans operate. Much effort has been put into determining what intelligence is, what behavior is considered intelligent, what is natural, and what is artificial. A vocabulary lesson precedes a discussion of the raw basis of expert system design. The reader is introduced to AI terminology: *chunk*, *long-term memory*, *cognitive*. This section includes discussions on ill-formed problems, varieties of knowledge, and the nature of expertise, each of which reads quite easily. Readers should give this early part of the book their complete attention; in some ways, it contains the most important information.

The chapter on representing knowledge is conversational, but exacting, with its talk about construction data nodes and how they are connected by links into wonderfully cryptic things called *semantic networks*. A great deal of time is spent on triplet nodes, how objects can be related to each other under relational rules, some different kinds of rules, and larger object descriptions called *frames*. The authors discuss drawing inferences: uncertainty, backward and forward chaining, and traversing inference nets. A final look at MYCIN, this time with a more critical eye, completes the first section.

The second portion of the work is a collection of abstracts, separated by classification, on commercially available expert systems. Each contains an overview and several parallel sections for ease of comparison. Included is an excellent chapter that describes a computer with an embedded, layered expert system; it identifies each layer (using many diagrams) and describes in detail the commonly used languages and tools in expert system design and operation. The essential philosophical differences among LISP-, PROLOG-, and MYCIN-based knowledge bases are elucidated, using both general and specific examples. Harmon and King dip briefly into history to mention important details that were discovered as a result of the practical implementation of expert system design theory.

Section three, "Developing Expert Systems," provides a simple example that would be helpful to the manager of a development group; otherwise this chapter makes only a minor dent in a mountain of material. Of course, this is the nature of an introductory text. The second part, which is worthwhile reading for anyone, provides an idea of how large system construction is approached—it ends up being an example application of standard software/system design methodologies.

The last section of the book is spent discussing applications for expert systems, with special emphasis on training considerations. Much of it is concerned with the present and future market, future hardware developments, and how to set up as either a prepared consultant, development group manager, or an entrepreneur. Most of this information will be of only remote interest to a technically inclined college student.

Perhaps the book's weakest point is the section with the capsule reviews of current products. Some recommendation on the part of the authors would have been welcome—a subjective indication, for example, as to why a company might spend $60,000 on one of these systems. This is somewhat different than understanding *what* they do. The person who is shopping for an application in this area can hope only to find some initial guidance here, perhaps to develop a short list of possibilities. Certainly he would not make a decision on a five-figure software system based solely on the recommendation of one book. Practical computer books like this one tend to have short life expectancies; this edition's may be shorter yet because the list of commercial tools will become dated. The authors may have reasoned that adding opinion based on the current state of the art would limit the text further.

This writing manages to present rationally the basics of expert system design, which automatically qualifies it as recommended for a first reading on the subject for computer science students. Too many inept presentations of the same material can be found.

In addition, this work lists companies that have ongoing artificial intelligence projects and offers a very complete list of references. While definitely aimed at business interests, it does not venture beyond the essentials, and it is certainly not exhaustively complete, but it is enjoyable reading nevertheless.

—MICHAEL B. BENTLEY

TECH BOOK

Accessories/Supplies

FREE DISK JACKET SAMPLE
Why settle for a paper sleeve for your floppy disks? The new low-cost DISKAT™ FLOPPY DISK JACKET gives superior disk protection and a practical way to print and display directory data. Manufactured from heavy duty antistatic vinyl, the DISKAT™ dual pocket design offers one pocket to protect disk from contaminants and physical damage, another to display formatted printout of directory data, free space, date/title info. Send for a free DISKAT™ sample and offer of free Directory Printout Software.
LEINBACH INCORPORATED
PO Box 730 Dept TJ1
Nyack, NY 10960
(914) 353-0990

PC CLEANER
Keep expensive equipment clean and static-free. Work on all automated office equipment. Removes dirt, grime, fingerprints, coffee stains, etc. Removes static. Easy to use. Also available; PC Screen Cleaner. $9.95 per 500ml bottle. $49.95 for PC six pack. Dealers inquiries encouraged.
Pimidon Corporation
47 Clarence St. Suite 444
Ottawa, Ontario, Canada K1N 9T1
(613) 236-9633

Bus. Opportunities

EXTRA INCOME
.. Learn 3 steps to Profitable Consulting
.... How to Locate Prospects
.... How to Bid on Projects
.... How to Reference Sell New Clients
.. Discover 4 Profitable Client Services
Invest $9.95 in "Make Your PC Profitable."
FREE Table of Contents sent on request.
TULSA COMPUTER CONSORTIUM (TCC)
P.O. Box 707
Owasso, OK 74055
918-747-0151

Personal Computer Owners
CAN EARN $1,000 TO $5,000
monthly selling simple services performed by their computer. Work at home-in spare time. Get free list of 100 best services to offer. Write:
C.I.L.B.Q.
PO Box 60369
San Diego, CA 92106-8369

Diskette Copy Service

BLANKET SERVICES
• Diskette duplication • Stocking/Drop shipping • Packaging • Fulfillment • 48 hour delivery. Place a blanket order with releases as you need them for any quantity at a fixed price. Prolok or Superlok • Lifetime guarantee. • No mastering fee • No chg for standard labels. Print no more sleeves than you actually need.
Star-Byte, Inc.
2564 Industry Lane
Norristown, PA 19403
(800) 243-1515 (215) 539-4300

Hardware/Add-on Boards

$200 PER MEGABYTE
Memory expansion for your IBM PC, XT, OR AT at $200 per megabyte. All the memory you've dreamed of at a price you can live with. JRAM-2 with JRAM-AT hold up to 2 megabytes per board with Ser/Par/Clks also available. Mega-memory and I/O ports all in one slot!

Tall Tree Systems
1120 San Antonio Road
Palo Alto, CA 94303
(415) 964-1980

SPEECH SYNTHESIS
SynPhonix: TRUE Unlimited Speech Synthesizer for IBM-PC/XT/AT/jr & compatibles. This low power short card includes an SSi263 speech chip, amplifier and speaker. Software includes Text-to-Speech, Phonetic Editor, Talking Clock & demos. Can be programmed with BASIC and other languages. Prices start below $200.

SynPhonix™
Electronic Speech Articulator

Artic Technologies
2234 Star Court
Auburn Heights, Michigan 48057
(313) 852-8344

LOW COST MODULAR DAS SYSTEM
Now available MODULAR INSTRUMENTATION DATA ACQUISITION SYSTEM (MIDAS) for IBM PC family. Hardware provides direct connection to standard transducers. SIGNAL CONDITIONING INCLUDED. Software provides PC DOS drivers for compatibility with other application programs. Features include low unit cost, high accuracy, modularity, and expandibility.

ADVANCED PERIPHERALS INCORPORATED
12650 West Geauga Plaza
Chesterland, OH 44026
(216) 729-3927

IEEE 488 CONTROLLERS
The most respected name in GPIB interfaces for microcomputers offers you two IEEE 488 Controllers for IBM compatible Personal Computers. Ziatech's experience guarantees you the best documentation and most efficient software available. Two-year warranty. Call or write for free information packet.

ZIATECH CORPORATION
3433 Roberto Court
San Luis Obispo,CA 93401 USA
(805) 541-0488 ITT Telex: 4992316

A/D TO RS232 CONVERTER
Low cost stand-alone data acquisition board for any 8 bit RS 232 port. 8 channels, 8 bit conversion. 0-5V input, 300-9600 baud. Small size, 4" × 5", with wall plug power supply. Easy-to-use, includes sample driven software listings. Kit w/all parts & PS, $59.95 + $4 shipping. Assembled & tested is $89.95 + $4.
TPS SYSTEMS
14820 Elmore Road
Anchorage, AK 99516
(907) 345-6730

SCSI-PC HOST ADAPTER
Plug & Play installation for SCSI Optical/Hard disk and tape peripherals. Wide selection of SCSI peripherals selectable by manufacturer and model No. via menu window. Absolutely NO driver software need be written by the user. Price is $249. for board, SCSI bios prom and floppy based configuration data.
Advanced Storage Concepts, Inc.
9660 Hillcroft #325
Houston, TX 77096
1-800-423-9175 (713) 729-6388

PC SPEED UP
10%-40% speed improvement for PC and clones. 100% compatible, easy 10 minute installations. Kit includes NEC V20 or V30 Chip, Tool User Guide, Warranty, free Multiuser BBS Access. V20-5 mhz for regular PC/XT $24.95. V20-8 or V30 replaces 8086. MC/VISA/CHECK. Tech info 414-242-2165.
Exec-PC, Inc.
P.O. Box 11191
Shorewood, WI 53211
Orders: 414-242-2173
Modem order: 414-964-5160

QUICK BOOT YOUR PC!!
For IBM PC's ONLY!! The PC-INMATE, a Firmware ROM Module for IBM-PC!! Features: BOOT within 10 SECONDS!!! BREAK THE 640K BARRIER!! Access up to 704K-of RAM! (LOTUS, DOS, FW etc.) SCREEN RELIEF!—protects your monitor, and MORE!!! Easily PLUGS IN-NO SLOT! Special Introductory Offer: $49.95 s/h incl. Dir inquiries accepted.
JS & J Software
1281 S. King St. Suite 6A
Honolulu, Hawaii 96814
Ch/MO/VISA/MC

Hardware/Cooling Devices

LESS NOISE! MORE COOLING!
Does your PC sound like a high speed turbine? Does it use expansion cards that create extra heat? Solve your noise & cooling problems with the SILENCER from PC Cooling Systems. Each model utilities a super quiet, West German, all metal, 4½" fan. Model SQ cools with 84% less noise than the standard fan it replaces! Model HP provides over 100% more cooling! Mounts to existing holes on the back of the IBM PC or XT. Immed. shipping. $79.95 + $4 s/h VISA/MC/CK.
PC COOLING SYSTEMS
31510 Via Ararat Dr.
Bonsall, CA 92003
(619) 723-9513

Hardware/Peripherals

33% FASTER AT—$19.95
If you have a PC AT you can plug in a new crystal, and watch it run 1/3 faster. Non-technical 10 minute installation (no soldering), and no expansion slot usage. Compatible with all software. Satisfaction guaranteed. Check or Money order. COD. $3 extra. VISA/MC
AVTEK ENGINEERING
P.O. Box 21-171
Eagan, MN 55121
(612) 452-7415

Hardware/Disk Drives

1.2MB PC-AT DISK DRIVES
AT your PC by replacing your diskette controller and adding a high-density disk drive. JDISKETTE handles 1 to 4 diskette drives plus and
I/O module. $249 for JDISKETTE-SA, $250 for TEAC 55GF (1.2MB); $200 for TEAC 55B (36-K); $180 for Ser/Par/Clk module.

TALL TREE SYSTEMS
1120 San Antonio Rd
Palo Alto, CA 94303
(415) 964-1980

Rates and Information

Standard listings consist of a bold lead line (23 characters maximum); 7 lines of body copy (45 characters per line); 4 lines for company name, address and telephone number. $110 per insertion—3 issue minimum. Additional charge for extra lines and company logos. Prepayment and frequency discounts available. American Express, MasterCard, Visa accepted. Copy subject to publishers approval. Send typewritten or printed copy, reproducible logo art (if applicable) and remittance to Kathryn Cumberlander, Classified Sales Manager, Ziff-Davis Publishing Company, One Park Avenue, New York, NY 10016. For additional information, assistance, or to place an order by phone, call collect (212) 503-5115.

TECH BOOK

8 INCH DISKETTE SYSTEM
Read, write and format diskettes from IBM mainframes, minicomputers, data entry equipment, etc. Complete easy to use software handles EBCDIC conversion. Can read and write CP/M 8-inch diskettes (many formats). You also can use 8-inch drives for PCDOS files; 1200 KB per diskette! $1150 complete.
MicroTech Exports
644 Emerson St. Suite 8
Palo Alto, CA 94301
(415) 324-9114

SOFTWARE/BAR CODING

BAR CODE-PRINT/READ
PRINT Bar Codes on PC and Epson/Okidata. Code 39 for $49. Code39, 2of5, UPC-A, & Codabar with disk data—$129. Subroutines—$129. AIAG-$299. LOGMARS, HIBC—$199. Read above bar codes thru keyboard attachment into any program (Wdstr, Lotus) on PC, XT, AT: *no* slot, serial port, or power—$595, (includes Code 39 print program).
Data Solutions Systems
130 Crespi Court
Santa Cruz, CA 95060
408/458-9938 Visa/MC

SOFTWARE/BUSINESS OPPORTUNITIES

RETRACTION
Below is the CORRECTED version of an advertisement which appeared in our May June July issues

SOFTWARE AUTHORS
WE WILL CONNECT YOU TO MAJOR SOFTWARE PUBLISHERS FREE. We are in contact with many publishers looking for marketable software. If you have a quality software package or program and wish to have it published and distributed, contact:
SOFTSEARCH, Inc.
P.O. Box 281
Budd Lake, NJ 07828
(201) 627-1790

WANTED OCR PACKAGE
TO BE DEVELOPED WITH NEW LOW COST OPTICAL SCANNER RETAILS FOR $1,195. Resolution 200 by 200 dots to the inch. Capable of running on IBM-PC and interfacing with IBM's display writer for the PC or other OCR applications. Call or write Mr. Mike Thaler;
IMAGE COMMUNICATIONS, INC.
511 Danbury Rd.
New Milford, CT 06776
(203) 355-3747

SOFTWARE/COMPILER

DeSmet-C - $109
Full K&R C compiler, linker, librarian, fullscreen editor, profiler, overlays, example software, 140 page manual. BOTH 8087 and floating point libraries. OUTSTANDING PRICE/PERFORMANCE. Consistently rated 1st or 2nd in published benchmarks. PC/MSDOS. No royalties on generated code. Newsletter. Updates $20 ea. $109 shipping included. Symbolic (C source level). Debugger Option $50 extra. C Compiler for the Macintosh $150.
C WARE CORPORATION
Dept. PCTJ
P.O. Box C
Sunnyvale, CA 94087
(408) 720-9696

C LIBRARY + WINDOWS + BTREE
******* BEST YOU CAN GET ********
**** C POWER FUNCTION LIBRARY ****
Over 325 richly commented tested functions
ALL SOURCE CODE! .... NO ROYALTIES!
**** 4 DISKS: 400 PAGE MANUAL *****
51 screen handling/graphic funct; 50 cursor/keyboard/data input; 85 superior string functions; 28 system status & control; 70 utility/DOS/BIOS/time/date; 42 printer control.
No matter what ELSE you have, get this one
Lattice, Microsoft, CI86, Aztec any full K&R
COMPLETE LIBRARY $129.95

******* C POWER WINDOWS ********
PROFESSIONAL WINDOW MANAGEMENT
overlays, borders, popup menus/help windows
file-to-window & keyboard-to-window support
monochrome or color, three kinds of menus
status-line, color, highlighting, much more
C WINDOWS: Complete source code... $129.95

******* SUPERFONTS FOR C ********
Extra size dramatic on-screen character fonts
Monochrome or color! Use our fonts or create your own ... Functions + font libraries .. $49.95

** B-TREE LIBRARY and ISAM DRIVER ***
Fast easy indexing! 16 million each: keys and records; variable length records! Multiple indices; examples; Full source code! PC-DOS, UNIX, XENIX, CP/M. Library only $79.95
ISAM ... $49.95, BOTH for only $109.95

Make utility: save time avoid errors $59.95

C WINDOWS + C LIBRARY all for $179.95
add SUPER FONTS FOR C only ... $199.95
add B-TREE & ISAM ($440 value) ... $299.95

Entelekon
SOFTWARE SYSTEMS

Entelekon
12118 Kimberley
Houston, TX 77024
(713) 468-4412

SOFTWARE/DATA ENTRY

Data Entry System
Heads down data entry for the PC will allow fast and accurate preparation of data files for your mainframe. Features: Auto dup, verification, auto format change, table look-ups, range checks, batch totals, dup key, record insert, and much more. Ask us about our free 30 day trial period. Only $395.
COMPUTER KEYES
6519 193 SW
Lynnwood, WA 98036
(206) 776-6443

SOFTWARE/DEVELOPMENT TOOLS

TURBO-PLUS™
Turbo-Plus™ is a set of external procedures crafted in assembler language and designed specifically for Turbo Pascal's interactive environment. Turbo-Plus™ includes: File Handle Disk Access Method, Instant Memory Mapped Text Display, Expanded Text Display. $34.95 plus $5 s&h.
NOSTRADAMUS
5320 South 900 East
Salt Lake City, UT 84121
(801) 261-0769

COLOR, CURSOR CONTROL
DATE/TIME, KEYBOARD, PEEK, POKE, COMMAND LINE ACCESS, and more available for IBM/MS FORTRAN, IBM/MS PASCAL, and GENERIC (MS) BASIC on PC/DOS and many MS/DOS machines. No redistribution fees. $47 U.S. MN residents add 6% sales tax. BARTSUBS by BARTSOFT
3210 Lake Shore Boulevard
Wayzata, MN 55391
(612) 473-4032

REVISION CONTROL SYSTEM
The Software Revision Management System™ stores all versions of source/documentation in a single ASCII file. Allows retrieval of any version of source and application of changes, while recording when, why and where changes were made with no duplication of common code. MS/PC-DOS 2.0 $99.95+. $5 s/h. (MN + 6%.) MC/Visa.
QUILT™ COMPUTING
7048 Stratford Rd.
Woodbury, Minnesota 55125
(612) 739-4650

CROSSREF/BASIC & PASCAL
Debug programs, customise listings, control evasive variables! Our utilities will list your source, cross-reference variables in sorted order, and list procedure/function names. Available for Microsoft® Basic and Turbo Pascal® at $29.95 each or receive both for $49.95. All orders add $4.00 S&H. Check or Money Orders (Inquiries invited) to:
EXECUTIL
PO Box ### (12516)
Albany, N.Y. 12212

** PASCAL'S FRIEND **
PASCAL'S FRIEND v2 contains source code for use with IBM PC Turbo Pascal: 1-2-3 style menu routines, keyboard handling, save & restore screens, read disk directory or any track & sector, system clock and calendar routines, write strings in any attribute, DOS function calls, MORE! $49.95 check, MO, Visa, MC
J.S. COMPUTING
815 N. 12th St. Suite 5
Allentown, PA 18102
(215) 821-9020

SOURCE DEBUGGER FOR LATTICE C
The MSD DeBugger™ is the last, and perhaps final, word in programming assistance for Lattice C users. C Debugger produces a high-level view of C programs via function names line numbers, variable names and C data types, plus a low-level view of machine addresses and instructions for testing assembler language functions. More features include:
- All documentation is prepared for programmers.
- Online help screen throughout the process.
- Capability to single step through your program.
- Set break points, examine registers and variables.

$165.00 + $3.50 shipping VISA/MC

Cdebugger
a high level language debugger

MSD
214 1/2 West Main St.
St. Charles, IL 60174
(312) 377-5151

C-INDEX™
C-INDEX is a state-of-the-art data management library for C. Ideal for data and text applciations. B+Tree, variable length records, multiple keys. We accept VISA, MC, Amer. Express cards. Money-Back Guarantee. Call or write for our free info packet.
- C-INDEX/FILE $99 Object Code
- C-INDEX/PRO $199 No Royalties
- C-INDEX/PLUS $395 Full Source

TRIO SYSTEMS
2210 Wilshire Blvd. Suite 289
Santa Monica, CA 90403
213/394-0796

PASCAL-TO-C-TRANSLATOR
Convert UCSD Pascal, MT+Pascal, and others to K&R C. Handles nested procedures, intrinsic functions, separately compiled units and modules, all data types including long integers. Requires 256K IBM PC/XT/AT. Send us up to 500 lines of Pascal and we will convert it for you FREE. Licensed source and executable code $5,000, executable only $2,500, or conversions 25 cents/line.

TGL
INC

TGL INC.
4400 Sulphur Springs Road
Corvallis, OR 97330
(503) 745-7476

TECH BOOK

CONTEXT-SENSITIVE HELP
TRUE/Helper adds context-sensitive help to existing programs. No access to program source is necessary. Works as a pop-up under the target program. Help text can be built using any editor. Text is compressed to reduce space. Works with most custom and off-the-shelf software. $39.95
GRW Software
P.O. Box 440007
Houston, Texas 77244
(713) 556-9878

ROMable CODE on PC!
PCLOCATE allows PC users to develop ROM-based software from MS-DOS "Exe" files. The user specifies the physical location of all segments. Output files are compatible with most PROM programmers. PCLOCATE supports the 8086, 8088, 80186, 80188, and 80286 processors. MC/VISA.
ALDIA SYSTEMS INCORPORATED
P.O. Box 37634
Phoenix, AZ 85069
(602) 866-1786

SOURCE CODE LIBRARY
.. "Structuring BASIC Programs" comes with:
.... BASIC examples on DOS 2.0. Diskette.
...... BASIC Check Writing Program.
.... BASIC Source Code Library
.... HOW to Structure Package only $9.95.
FREE Fact Sheets.
FREE Contract Programmer Registration.
TULSA COMPUTER CONSORTIUM (TCC)
P.O. Box 707
Owasso, OK 74055
(918) 747-0151

TURBO PROFESSIONAL™
Turbo Pascal Tools
*SERVICE INTERRUPTS IN PASCAL
*MAKE YOUR ROUTINES RESIDENT
*VARIABLE SIZE WINDOWS
*KEYBOARD MACROS
*DOS PROGRAM EXECUTION & RETURN
*SOURCE CODE INCLUDED
*11 INCLUDE FILES
*109 ROUTINES
*SUPER MACS EXAMPLE PROGRAM
Super Macs features: define macros, save macros, load macros, movable window, prints concurrently in DOS 3. Works within other programs. Available separately w/o source code.

SUPER MACS ONLY (.COM)	—$10.00
WITH TURBO PROFESSIONAL & SUPER MACS SOURCE CODE	—$49.95
SHIPPING & HANDLING	—$ 5.00

SUNNY HILL SOFTWARE
13732 Midvale North, Suite 206
Seattle, WA 98133
(206) 367-0650
Requires 100% compatible, DOS 2-3, Turbo Pascal 2-3 for compatibles.

TURBO EXECUTIVE™
For Real-Time Multi-tasking, control applications. Interrupt driven, Turbo Pascal source code includes a background ASYNC COM task, memory resident code example, windowing, video and directory utilities!
- Non Pre-emptive version $64.95
- Pre-emptive (time slice) version $164.95
- Call or write for FREE brochure consulting and programming service available.

Clear Creek Computing
2107 Audubon
League City, TX 77573
(713) 332-9940

X-VIEW 86™
Analyze execution of DOS software. Find code hotspots, identify compatibility problems, find bugs beyond DEBUG's reach. You can breakpoint programs on I/O or INT instruction execution or on memory references. Requires IBM PC DOS DEBUG 2.0 or 2.1 with PC/MS-DOS 2.0 or greater. $59.95 plus $2 S/H. Texas residents add $3.67 ST. Check, MC, Visa, AE, DC.
MCGRAW-HILL INC.
8111 LBJ Freeway
Dallas, Texas 75251
1-800-221-VIEW (in Texas 1-800-233-VIEW)

BASIC DEVELOPMENT TOOLS
The Programmer's Tool Kit is a collection of eight preprocessors and utility programs that speed development, documentation, and debugging. Code using labels instead of line numbers, remove line numbers, merge, cross-reference, compare, list, and compress. Write or call for full details.
DATASMITH, INC.
Box 8036
Shawnee Mission, KS 66208
Phone: (913) 381-9118

BTRIEVE/BETTERBASIC XFACE
*Do ALL Btrieve operations from BetterBasic
*Written in Assembler: COMPACT, FAST, SOLID
*Use via CALL, as a Procedure or as a MODULE
*FULL documentation, example programs, files
*Conforms to ALL Btrieve/BetterBasic specs
*FREE UPDATES for first six months
*IBM-PC/XT/AT DOS 2.x/3.x *$49.95 MC/VISA
JOSEPH A. VINCENT CONSULTING
805 Pine Way
Anchorage, KY 40223
(502) 580-2917

MATIS/T FOR TURBO: $29.95
Create large virtual screens; formatted input field; windows for single or multiple screen display. Total control of video attributes. Memory resident Assembler routines used as PROCEDURES in TURBO PASCAL. Also available for BASIC, C, MS-PASCAL, & Assembler: MATIS $49.95. Royalty FREE. Add $5 s/h. VISA/MC. CA add tax.
SOFTWAY, INC.
500 Sutter St. Suite 222TL
San Francisco, CA 94102
(415) 397-4666

C/FORTRAN EFFICIENCY PACKS
The efficiency packs contain tools to assist in code management, configuration management, execution speed-up, documentation and symbol cross-reference, plus a number of utilities. Prices are $465 for C and $275 for FORTRAN. Individual tools may be bought separately. Call-toll-free for technical descriptions.
FIGUREFLOW LTD.
9, Market place, Hadleigh
Suffolk, United Kingdom IP7 5DL
1-800-362-3625

REALIA COBOL SITES
REALFORM is the Screen Handler you have been looking for. It features interactive screen design, an on-line forms editor, relief of cursor control, expedites screen generation, supports numerous data field types, is easy to use and offers powerful data entry functionality. $395. US.
CHARANDON COMPUTER SYSTEMS INC.
1600 Laperriere Avenue
Ottawa, Ontario, Canada K1Z 8P5
(613) 729-3003

COBOL SCREENS-FORMMAKER
IBM PC or compatibles. Interactive Screen design utility for MS/PC COBOL or CIS/Level II COBOL. Save time—clean screens in minutes, not hours. Features: Maintainable screen files; user assigned field names; auto generation of Screen Section and/or W/S copy files; much more. Specify Complier type. $95.
Logicware Data Systems, Inc.
Box 9C Sunnybrook Road
Jackson, NJ 08527

TURBO LIBRARY
Cut development time. Use Input, windows, quick screen output, screen save and restore, menus, fill-in templates, template editor, reports and string utilities. For IBM PCjr, PC, XT, AT with Turbo Pascal V2.0 and higher. Money back if not satisfied. Documented source code on disk $49.95.
PLATTWARE
P.O. Box 2626
Del Mar, CA 92014

BETTER TOOLS FOR BETTERBASIC
BetterTOOLS,™ 100 procedures & functions that speed BetterBASIC™ development. Sorting, scrolling, extended math functions, video routines, disk directories, formatted screen display, on-line error descr., input w/full editing & extended code trapping, much more. BetterTOOLS™ w/source & manual, $89. Write:
Software Associates
6220 W. Airport Blvd.
Houston, TX 77035
(713) 726-0706

INSIDE—DISASSEMBLER
Get INSIDE your PC, XT, and AT software with the disassembler for the 8088/8086, 80186, 80286, 8087 & 80287. INSIDE generates address labels, listings and X-REF table. Write code to disk in "macroassembler" form. Command driven; simple to use. INSIDE and manual only $49.95 plus $3.00 s/h (IA res. add 4% tax.)

Inside DisAssembler

BLACKRIDGE CORPORATION
P.O. Box 385
Betterndorf, IA 52722
(319) 355-4465

TOOLS FOR CB80 & CB86
BDOS, DOS, and BIOS calls from CB80 and CB86! CBC Tools includes functions for directory access, string ops, a debugger, radix conversion, command line parsing, quicksorts, bit and byte ops, and much more. Available for CP/M-80, CP/M-86, and PC-DOS for $180.00.
Minnow Bear Computers
P.O. Box 2233 Sta. A
Champaign, IL 61820-8233
(217) 398-6883

BTREES IN C SOURCE
BTree
- fast and portable
- duplicate keys
- variable length records

ISAM
- works with BTree
- simple, powerful application file system.

Snake
- Similar to UNIX make
- supports nested macro, file name expansion.

All three
ALL SOURCE CODE INCLUDED, NO ROYALTY FEES.

SOFTFOCUS

Softfocus
1343 Standbury Drive
Oakville, Ontario, Canada L6L 2J5
416-825-0903

DISK DRIVE DIAGNOSTIC
Memory Minder, from J & M Systems, is a disk diagnostic program for the IBM PC, PCjr, & IBM compatibles. It checks your drives for head alignment, spindle speed, hysteresis, azimuth & more. And, you can use Memory Minder to actually align your disk drives! $99 plus $4 shipping.

J & M SYSTEMS, LTD.

J & M Systems, Ltd.
15100-A Central SE
Albuquerque, NM 87123
(605) 292-4182

TECH BOOK

SOFTWARE/EDIT

PQfEDIT
Full-screen editor, EDLIN alternative for the PC family. With: 4-way cursor, insert/delete/replace char, 4-way scroll, split/join lines, add/copy/move/delete lines, search/replace strings, get/put to disk, help screens. Edit multiple files at same time. Needs 128K and MS-DOS. $45 (+$1.80 tax in VA) + $2 shipping.
ADAM Systems Corporation
5919 Munson Court, Dept. 404
Falls Church, VA 22041
(703) 379-0669

LIL' JAKE
So long EDLIN! Hello Lil' Jake!
A Command-Free Full Screen Editor for IBM PCs and compatibles. Comprehensive on-line help facility; global searches/changes, file sorts, split/join; editing several files simultaneously; DOS commands within editor; user customization; and more. $49.95 + S&H.
Fooks Mathewson Company
P.O. Box 2424 (Oper-54)
Arlington, Virginia 22202
(703) 684-8284

SOFTWARE/ENGINEERING

smARTWORK®
Printed-circuit artwork editor for double-sided boards up to 10" by 16", runs on an IBM-PC. Color display allows complete interactive control over the placement and routing process. 2X artwork can be made on a dot-matrix printer or pen-and-ink plotter, $895. Write or call for a brochure.

WINTEK

WINTEK Corporation
1801 South Street
Lafayette, IN 47904-2993
(317) 742-8428

METAL FABRICATORS
PC/Cutlist takes input from your bill of material—Detail drawing and calculates the best cutting combination for any length stock and prints a shop ready cutting list and scrap report. Also an optimization feature finds best mult length for mill orders. Price $300. Demo Disk $25.00
THE JOSEPH ALBERT CO.
P.O. Box 611
Blue Island, Illinois 60406
(312) 349-9032

STEAM
STEAM computes the thermodynamic properties of water over a wide range of values commonly used in applied thermodynamics. The FORTRAN package includes an interactive "stand-alone" program and code in object form, callable as a subroutine. Specify your FORTRAN compiler. $295. (Calif. + Tax).
SOFTset, INC.
10889 Wilshire Boulevard, Suite 810
Los Angeles, CA 90024
(213) 824-1904

CREATE PCB's EASILY
Easily create double-sided printed circuit board artwork of up to 21" square using PC-Layout on an IBM-PC. the artwork is created using a color artwork editor. the final artwork, solder mask and silk-screen (text) is produced using either a printer or a plotter. Price $745. Demo disk/manual $45.
Draco Technology
7210 Jordan Ave., Suite D50
Canoga Park, CA 91303
(800) 235-6646 ext. 646 (outside CA)
(800) 235-6647 ext. 646 (CA only)

SOFTWARE/FINANCIAL

CASH FLOW MANAGER
Easy, powerful system for personal budgeting, bill paying, smart checkbook management, and cash-flow control for IBM PC and true compatibles. Take charge of your money with CashFlow Manager! 128K, one DSDD floppy, PC DOS 2.x; printer optional. $39.95 check, Visa, MC. Tax Deductible. (add 5½% in OH)
SINGULAR SYSTEMS
4250 Possum Run Road
Dayton, OH 45440

SOFTWARE/GENERAL

PUBLIC DOMAIN SOFTWARE INC
Over 60 volumes of public domain software in CP/M & MS-DOS formats.
- editors & compilers
- text formatters
- communications packages
- many UNIX-like tools

Write or call for more details.

Users' Group

THE USERS' GROUP
P.O. Box 97
McPherson, KS 67460
(316) 241-1065

SOFTWARE/GRAPHICS

TEKTRONIX 4010 EMULATION
High resolution screen/printer graphics for the IBM PC. Full interactive capabilities with file transfer and cross-hair control. Utilizes PLOT 10, protocols. OFF-line review of graphics output. Supports IBM color/graphic or Hercules cards. Easy to use 4010 emulation at an affordable price. $80.00. Demo disk $5.
Technological Systems Group
5044 Haley Court
Lilburn, GA 30247
(404) 923-4980

FOR PRO & NOVICE ALIKE
BOARDS: Tecmar, Hercules, IBM E/CGA, AT&T. LANGUAGES: DeSmet-C, C86, Lattice-C, MS-C, Turbo Pascal, MS-Pascal, MS-Fortran, MS-Macro86. PRINTERS: Epson, Okidata, Toshiba. FAST: 200k pixels/sec avg. TIGHT: 10K. COMPLETE: 2/3D-Lines, Bit Blks, Windows, Color, Fill, Arc, Icon, Zoom, Trig, Character, String, Mouse, Keyboard, Timers, Menu Grids, & more. PLUS: Icon Editor, Fonts. No royalties! MODULES: $35.

SoftRoms™ High Performance Graphics Tools

AFFINITY, INC.
126 Normal Ave.
Buffalo, New York 14213
(716) 882-5077

PRESENTATION GRAPHICS
Create transparencies, 35mm slides, printer and plotter output with Sound Presentations. Construct organization and flowcharts, maps, and floor plans using Sound Presentations' comprehensive libraries and powerful editing features. Enhance and modify Lotus charts and graphs. $250 + $50 for Lotus convertor.
Communication Dynamics, Inc.
8532 SW St. Helens
Wilsonville, OR 97070
(503) 682-0651

GRAPHICS LIB FOR TECMAR
TEK-MAR lets you do high-res graphics on your TECMAR Graphics Master. TEK-MAR is a library for use with MS Fortran. Features windowing, viewporting, clipping, axis rotation. Similar to Tektronix graphics. Includes screen dump/restore, Epson screen print, support for Hewlett Packard, Western Graphtec plotter. Requires MS-FORT 3.20, 320K, GMDEV.SYS (5.2). Price: $195.
ADVANCED SYSTEMS CONSULTANTS
18653 Ventura Boulevard, Suite 351
Tarzana, California 91356
(818) 407-1059

SuperCAD a2.5D system.
Developed by a Columbia University graduate, it's very fast and powerful, supporting all the most advanced I/O devices and the 8087. It works like a screen oriented word processor for graphic primitives and drawings. Free 1 yr. update service available in the US. Send $1500 by check, L/C or P/O. Dealers Welcome!
CAD-CAM-II SRL
Via Fornaci 1
06034 Foligno ITALY
Tel: (0742) 20921

SOFTWARE/LANGUAGES

EASY WAY TO MASTER DOS!
New handbook shows how, why, and when to use all 44 PC & MS DOS commands quickly and easily! Lively hands-on tutorial makes learning fun, for a change. Covers basic thru advanced topics. All versions up to 3.1. Lots of practical examples. 221 pages only $15.95 at bookstores, or send check + $1.00 S&H to:
LITTLE, BROWN & COMPANY PUBLISHERS
200 West Street, Code T-368121
Waltham, MA 02254
1-800-343-9204 (credit cards)

ASSEMBLY PROGRAMMERS
Your library is here. Graphics, Floating Point Math, Trigonometry Functions, Windows, 8087 Support, plus much more. Supplied on IBM PC-DOS diskettes for only $49.95. Source code available for an additional $99.05. Written in MS Assembly Language for DOS 2.x or newer. Get yours today!!
BC Associates
13073 Springdale St., Suite 134
Westminister, CA 92683

CCSM—LOW COST PC MUMPS
The most efficient productive programming language is now available for only $59.95. Graphics add-on is $49.95. Features include Full Screen Editor, Virtual Memory, 8087 and BCD Support, Transportable Micro-Mini-Main. Productivity can be increased as much as 300% over other languages. Fully documented.
GUIDANCE SOFTWARE
P.O. Box 5362
Kingwood, TX 77325
(800) 257-8052

muLISP-85™ RELEASED
muLISP-85 is an integrated AI programming environment that includes over 350 Common LISP functions, a high performance compiler and interpreter, a window-oriented source file editor and debugger, an on-line tutorial system, and a detailed reference manual. Write or call for more detailed information.
SOFT WAREHOUSE, INC.
3615 Harding Avenue, Suite 505
Honolulu, HI 96816
(808) 734-5801

SOFTWARE/OPERATING SYSTEMS

pcSHARE MULTI-USER O/S
pcSHARE allows your IBM-XT or compatible to support up to 4 users running 123, dBASE, WordStar, etc on inexpensive serial CRTs. For the software developer, pcSHARE efficiently runs compiled Basic, Pascal & C programs with full DOS 3.0 record locking. No risk 30 day money back guarantee.
DIGITROL COMPUTERS, INC.
440 Phillip Street
Waterloo, Ontario, Canada N2L 5R9
(519) 884-4541

TECH BOOK

SOFTWARE/PUBLIC DOMAIN

BEST OF PUBLIC DOMAIN
Fantastic Software at Unbelievable Prices!! We have put together 18 volumes of disk packages, all categorized and fully tested for easy selection. You may also order custom disks, with ONLY the software YOU need. Just $4.00 for a Sampler Disk with our complete catalog, or $1.50 for Catalog alone.
COMP-U-SOFT
2530 Berryessa Road, Suite 410
San Jose, CA 95132
(408) 272-9689

SOFTWARE/SCIENTIFIC

HP-PC HYPER-CALCULATOR
Activate emulation of an HP-11C programmable scientific calculator from any program, and return with the results. Features program and data storage on disk, indirect addressing, alpha-numeric messages, printing, on-line HELP, 100 registers, and 1,000-step programs for solving roots, numerical integration, complex numbers, matrices, base conversion, time/money problems. $39.95.
SUNDERLAND SOFTWARE ASSOCIATES
P.O. Box 7000-64
Redondo Beach, CA 90277
(213) 540-2567

SOFTWARE/SECURITY

SECURE AT/XT/PC
Control system access, data access! FiXT/S. Control system boot for most popular XT/PC hard disk controllers. Vfeature for AT-and-XT-compatible HD controllers segments hard disk by volumes, controls access with passwords, supports hard disk expansion. $70-$130 + $3 shpg. plus CA tax.

Golden Bow Systems
P.O. Box 3039
San Diego, CA 92103
(619) 298-9349

SIDEVIEW: PC/AT MENU
Password protect your hard disk today! DOS 3.0/3.1 hard disk manager. Allows you to create color menus to run any program or DOS command. Prevent hard disk erasure. Works with XT's too. 25 pg manual. Money back guarantee. PO's & phone orders accepted. $39 + $3 shpg.
Keller Software
1825 Westcliff Drive
Newport Beach, CA 92660
714-642-5544

SOFTWARE/SERVICES

TAPE TO DISK CONVERSIONS
Convert any 9 track magnetic tape to or from over 500 formats including 3½", 5¼", 8" disk formats and word processors. Formats available include IBM-PC, Apple, Altos, TRS 80 8" CP/M, Displayer., IBM Sys/??. Macintosh, Wang, Lanier, OS/6 and 500 more. Disk to Disk conversions also avail. Call for more info.
PIVAR COMPUTING SERVICES, INC.
47 W. Dundee Rd.
Wheeling, IL 60090
(312) 459-6010

DATA CONVERSION ON PREMISES
Keep your data confidential. TO/FROM cards, disks, tapes. ASCII/EBCDIC, 800/1600/3200BPI 9-Track. For IBM-PC/XT/AT and the compatibles. Hard disk and Bernoulli Box owners, we will bring our systems and do the conversion on your premises.
DP Modern Systems
804 North Reeder Avenue
Covina, CA 91724
(818) 966-2628

SOFTWARE/SPREADSHEET

FORMAT—1-2-3
- **Loads ANY ASCII data into Lotus™!**
- 1-2-3 and Symphony Utility!
- Send spreadsheets using E-Mail!
- On-Line Help System!
- Runs on PC, XT, AT, and lookalikes!
- Also TI Professional, AT&T
- Extended Warranty Available!
- DOS 2.0 + with 64K, 1 drive
- $150, MC, VISA, CK, COD

RELIABLE SOFTWARE COMPANY
P.O. Box D
Titusville, FL 32781
(305) 267-2043
(800) 235-6646 Ext 581 (nat)

SOFTWARE/STATISTICS

RATS!
RATS is the ultimate econometric software package. It is powerful, fast, accurate, and inexpensive. RATS performs regression analysis, time-series forecasting (including Box-Jenkins) and cross-section (including logit and probit). RATS supports a variety of data formats, including Lotus WKS files. RATS also offers a programming language that provides maximum flexibility and power. Spectral analysis and graphics are available as options. Base program $200. Demo including 300 page manual $40. MC/VISA.
VAR Econometrics
P.O. Box 19334
Minneapolis, MN 55419
(612) 822-9690

DEMOGRAPHIC DATABASE
NOTHING-BUT-THE-FACTS is a general purpose database on 32 diskettes that covers all the US countries and cities of 25,000 people or more. Includes data on agriculture, banking, crime, climate, education, government, housing, income, poverty, vital statistics, and much more. Manual, county map and storage cases included. Ideal for market research. $290 MC/VISA.
Melissa Data Co.
12 Balboa Coves
Newport Beach, CA 92663
(714) 650-1000

SOFTWARE/UTILITIES

NOBLINK™
ErGoNoMiCs! Eye strain? With just a keystroke you can change your blinking cursor into a friendly solid block reverse video cursor. If you wish you can change it back and forth (Blink/No blink) right in the middle of a program, such as your word processor. Give your eyes a rest. $29.95 plus $5 s&h.
NOSTRADAMUS
5320 South 900 East
Salt Lake City, UT 84121
(801) 261-0769

Hard Disk DIRECT ACCESS™
The ULTIMATE utility for any hard disk user. DIRECT ACCESS is a powerful hard disk management tool. Organizes your software programs into a "user defined" menu system. Features attractive screen display, rapid access into an application using a single keystroke & easy to learn. Also handles batch files & complicated DOS commands. Order today. Only $59.95 + $5.00 shipping & handling. MC/VISA.

DELTA TECHNOLOGY

DELTA TECHNOLOGY
P.O. Box 1104
Eau Claire, Wisconsin 54702
(715) 832-0958

AT/XT/PC HARD DISK EXPANSION
Replace hard disk with a bigger one, or add a second drive! Vfeature supports high capacity drives on standard AT, XT, and compatible hard disk controllers. Includes multiple volumes, security features, selectable clusters, keyboard lock. $80-$120 + $3 shipping + CA tax.

GOLDEN BOW SYSTEMS
P.O. BOX 3039
SAN DIEGO, CA 92103
(619) 298-9349

DISPLAY ANY MS-DOS FILE
List ASCII and binary files. TRUE/View can be invoked as a pop-up or directly from DOS. Switch between ASCII and hex modes. View multiple files. Search for string patterns. Mark file positions for easy retrieval. Make small changes with edit feature. User configurable. $19.95
GRW Software
P.O. Box 440007
Houston, Texas 77244
(713) 556-9878

TRUE PATH UTILITY
Never misplace files again. TRUE/Path extends the MS-DOS PATH utility to search for data files as well as programs. Unlike other path programs, TRUE/Path uses the same path given by the MS-DOS PATH command to locate data files. Works with all MS-DOS software. A perfect for inexperienced users. $19.95
GRW Software
P.O. Box 440007
Houston, Texas 77244
(713) 556-9878

CONVERT ANY ASCII FILE
The DataMessager-the ESSENTIAL ASCII file conversion utility. Handles up to 256 simultaneous string search/replace operations in any file plus two levels of character translation. Supports wildcards and smart buffering, too. Menu driven-fun to use, easy to learn. Unlimited applications in wordprocessor/printer command conversion, file interchange, communications. PC/PCjr/XT/AT w/192k, 80 col. Only $155 postpaid.
R.A. Handverger
281 Village Street
Medway, MA 02503
(617) 533-2302

EASY AIM-30 day trial
Installs your programs on a hard disk and creates subdirectories, paths, and displays programs in a menu, all automatically. Features include time log, backup to floppy or tape, copy files to/from floppies to/from/between menu selected directories. DOS is transparent to end user. Powerful yet simple. $84.95.
LARSON SYSTEMS, INC.
BOX 193
STORY CITY, IOWA 50248
(515) 733-2717

IBM USES CPM DISKS!!
Uniform lets you use CPM disks in your IBM B: drive while running any PCDOS/MSDOS program. Read and write to the CPM disk directly or transfer files to IBM format. Over 70 formats also 8" and 96 TPI, IBM PC, XT, AT. Call for details. Dealers invited. (Not for Apples). $59.95 + $3 S&H. Uniform by Micro Solutions.
BLUE HERON
1108 South Second St.
Dekalb, IL 60115
(815) 758-2355

TECH BOOK

UNP/UNPROTECT MS-BASIC
This utility may be used to unprotect programs written in MS-BASIC (MS/PC-DOS, BASIC, GW-BASIC, etc.), make your own modifications to standard programs or learn some professional programming techniques used in accounting and educational software. Send check or M.O. for $30 (NJ add $1.80 tax).
IRA YERMISH & ASSOCIATES
232 Society Hill Blvd.
Cherry Hill, NJ 08003
(609) 424-9448

DISK MECHANIC
THE ULTIMATE Floppy Disk Backup & Repair Utility. Can back up ALL software protected disks written on the IBM PC. Works manually or automatically. Files or sectors can be restored, searched, examined & changed. Checks disk drive speed, req. IBM PC, XT, AT, DOS 1, 2, 3, 192K + 64K if only 1 floppy drive. $73 ppd. USA
MLI MICROSYSTEMS
PO BOX 825, Dept TB2
Framingham, MA 01701 USA
(617) 926-2055 for info MC/VISA

DOS PATH COMMAND FOR DATA
Now Lotus, dBase, Wordstar, etc. can access data files no matter where they're located. DPATH + works like DOS PATH command-up to 128 dirs. can be searched. Handles output files, fileclosing and virtual disk too. Wildcards OK. Transparent. Requires only 4K. MS-DOS 2/3. $45 + $5 s/h MD. Add 5% VISA/MC/P.O./check.
Personal Business Solutions
P.O. Box 757 Dept. T
Frederick, MD 21701
(301) 865-3376

THE NIBBLER™ 2.0
Provides ultimate DATA INTEGRITY for DOS systems. Powerful, easy and up to 10X FASTER than NORTON! 2-key UNERASE. Edit-View sectors encrypt, wipe or copy files. SEARCH memory, files or disks. View and edit 1 MEG of memory. Supports IBM-AT, FIXED and RAM disks. DO3 3.1 $69.95+$2.50S&H. CA 6% CHK-MC DEALERS WELCOME!
TACHYON SYSTEMS INC.
2725 Congress Street. Suite 2H
San Diego, CA 92110
(619) 574-1666

PRINTERWINDOW™
Printer utilities for IBM PC & Compatibles.
- Pop-up memory resident printer control.
- Epson & IBM printer emulation.
- Multi-size screen dumps; text formatting.
- Font editor; Wordstar installation & more.

Packages available for NEC, CITOH, OKIDATA, EPSON and other popular dot matrix printers. Info & demo disks available. $49, VISA/MC accepted.
Courtrin Enterprises, Inc.
P.O. Box 231190
San Diego, CA 92123
(619) 569-8308

CHARACTER CUSTOMIZATION
CHARGENI works with the IBM EGA to let you modify the character set, allowing many word processors to display technical material, equations or other special characters. Also works on the Color/Graphics Adapter in graphics modes. Requres DOS 2.x or 3.x, IBM Standard or Enhanced Graphics Adapter. $25 + $2 s/h (MN add 6%).
DK Micro Consultants
P.O. Box 6714
Minneapolis, MN 55406
(612) 722-0931

DISK MANAGER
Why not be organized? LIBBY 2.0 is a full-featured disk librarian that can fully organize any disk library—consolidates hard/floppy directories into 1 master file. Professionally written, ver. 2.0 supports hard disk/DOS 2.0 & later/numerous report & search options. 128K, 2 drives. $29.95 USA.
ADVANCED PC SYSTEMS
3080 Summerhill Court
San Jose, CA 95148
(408) 270-6248 for info, VISA/MC

32-bit TURBO INTEGERS
We LOVE TURBO PASCAL. One SHORT complaint: NO LONG INTEGERS! We wrote a 32-bit integer package in ASSEMBLY LANGUAGE! It's SMALL, FAST, and includes RUN-TIME error CHECKING! First 100 orders get a million-record SORT package OR a $5 rebate (please specify when ordering). $24.95 + $1 s&h.
Cauthorn & Associates
1811 N. Kaspar
Arlington Heights, IL 60004
(312) 259-3930

EXPECT A MIRACLE $10
- Packed Utility Disk
- For IBM PC
- Hot Stuff!!

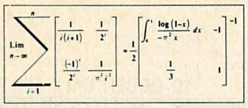
•••••••••••••• $10 ••••••••••••••

Packed Utility Disk

Micro Tools Inc.
P.O. Box 357
North Chili, NY 14514
(716) 594-1088

SOFTWARE/WORD PROCESSING

FORMATH™ TEXT-FORMATTER
For technical texts. Sizes & places equations, matrices, ratios, integrals, big symbols, footnotes. Macros, multiple fonts, 158 Greek/math symbols. Automatic hyphenation, section/equation/reference numbering, indexes, table of contents. Dot-matrix, daisywheel, laser printers, plain/graphics monitors. $400. $50 for demo.



SHANTHA SOFTWARE, INC.
50 West 97th St. Room 11N
New York City 10025
212-222-7647
Touch-tone Toll-free: 950-1088-wait-FORMATH

PC BUYERS GUIDE
THE INDEPENDENT GUIDE TO PRODUCTS AND SERVICES FOR IBM Personal Computers. This master directory lists, describes, rates for ease of use, and tells you about hte latest in software, hardware, accessories, supplies, books, services, et al. Everything for you and your IBM PC. It even includes an Index of Computer Camps for both adults and children. From the publishers of PC. An ideal gift at only $7.95. (+ $1.55 shipping/handling) to U.S. destinations; and $11.00 elsewhere. Residents of CA, CO, CT, DC, FL, IL, MA, MD, NJ, and NY add applicable sales tax. Checks of MasterCard or VISA acceptable (with number, expiration date and signature). U.S. funds only. Mail to:
PC BUYERS GUIDE
CN 1914
Morristown, NJ 07960

TECH MART

Sort RM/COBOL Files

• • • QISORTC • • •

Finally, a sort utility for RM/COBOL. Using efficient algorithms in assembly language making it the fastest sort on the MS-DOS market. Qisortc supports:

- RM/COBOL Version 1.5x & 2.xx files
  - Single / Dual File Indexed
  - Relative and Sequential
- ASCII, COMP-1, & COMP-3 fields
- 32 sort control fields
  - ascending / descending
- 32 record selection fields
  - EQ:NE:GT:GE:LT:LE field:constant
  - implied AND, OR may be specified
- 32 output fields or total record
- CALL from RM/COBOL, when provided subroutine is linked with runtime
- Use pathnames and return codes
- Keyboard or Parameter file
- Requires MS-DOS ver 2.xx or above
- Only $95.00 - COD, VISA, or M/C.

Quantumn Information Systems, Inc.
145 N.W. 85th St., Suite 103
Seattle, Washington 98117
(206) 789-2888 Telex 4941041

CIRCLE 387 ON READER SERVICE CARD

TERMINAL EMULATION

Softerm PC emulates over 30 popular terminals including the:

- DEC VT102, VT220
- Data General D200, D410
- IBM 3101-20 (block mode)
- Hewlett-Packard 2622A
- Honeywell VIP7801, VIP7803

Guaranteed Compatibility
Call for free product brief
$195 MC-VISA-COD
For the IBM PC/XT/AT, DG1, NEC, Wang PC, TI Pro, Gridcase, Tandy

SOFTRONICS
3639 New Getwell, Suite 10
Memphis, TN 38118
901-683-6850

CIRCLE 394 ON READER SERVICE CARD

THE HAMMER
Software Tools in C

"I have already saved weeks of coding . . . thank you for providing such a useful tool . . ." G.T.

Let **The HAMMER Library** of over 150 routines save you valuable development time and effort:

LIBRARY FUNCTIONS
- **Multi-level 123-like MENUS**
- **DATA ENTRY**
  - MULTI-FIELD mode for Full-Screen data entry
  - Single-Field mode for individual fields
  - Data Verification
  - Full Editing within each field
  - Strings, dates and fixed decimal numbers
  - "Option" fields force user to pick from a given set
- **SCREEN MANAGEMENT**
  - cursor positioning
  - display boxes & tables
  - full attribute control
  - scrolling and clearing
- Date/time/string conversions
- BIOS access/pattern matching/**and more**

SUPPORTS: Microsoft C 3.00 • CI-C86 • M W C86 • DeSmet C • Lattice

INCLUDES source code and manual — $195 plus shipping, VISA/MC accepted

O.E.S. SYSTEMS
1906 Brushcliff Rd. • Pittsburgh, PA 15221
412/243-7365

CIRCLE 388 ON READER SERVICE CARD

PC THE BUYERS GUIDE

THE MASTER DIRECTORY OF PRODUCTS FOR THE IBM PC, PC XT, PCjr AND MOST COMPATIBLES!

AVAILABLE AT YOUR LOCAL NEWSSTAND AND COMPUTER STORE.

FAST SCREEN OUTPUT FOR TURBO PASCAL

FASTSCREEN ™ is a set of **inline** assembler and Pascal procedures for Turbo Pascal users.

- Display an entire screen or window almost instantly
- Process multi-field input screens that give your user full cursor control
- Sample program uses Conway's **LIFE** to illustrate use of procedures
- All source code included
- IBM PC/XT/AT and true compatibles

29.95 — VA residents add 4% sales tax

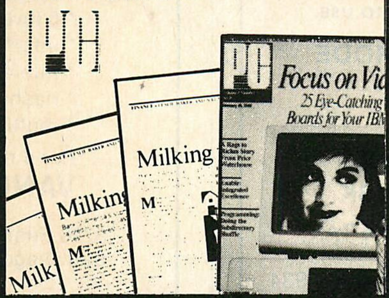
TECHNISOFT
1710 Allied Street, Suite 37
Charlottesville, VA 22901
(804) 979-6464

Turbo Pascal is a registered trademark of Borland International

CIRCLE 397 ON READER SERVICE CARD

REPRINTS AVAILABLE

Quantity reprints of articles appearing in PC Magazine are available and will be prepared to meet any special requirements. Inquiries should be directed to Eileen Pfeiffer, Reprints Dept., Ziff-Davis Publishing Co., 1 Park Ave., New York, New York 10016. Phone 212-503-5447.



Will you be making your next micro interconnect decision in the dark?

Shed light on the subject with

DATA DECISIONS
PC COMMUNICATIONS

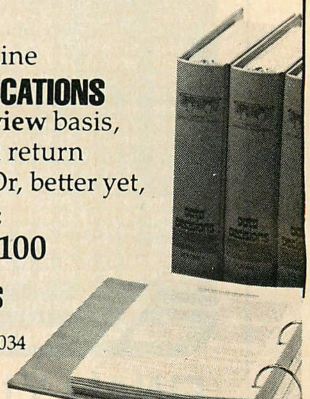
A unique new One-Stop Information Source that *tests* and *evaluates* all the hardware and software products you need to link your pc's to corporate mainframes, minis, data services, and other pc's.

It comes in 3 loose-leaf volumes, updated every month to ensure currentness and completeness.

Along with hands-on test results, you also get action-oriented technology updates, concise product profiles, reliable user ratings, monthly newsletters, a comprehensive vendor directory for the US and Canada... and *unlimited free telephone consulting by micro experts whose only job is to answer your questions.*

To learn how you can examine **PC COMMUNICATIONS** on a **Trial Review** basis, just fill in and return the coupon. Or, better yet,

Phone Today:
(609) 429-7100



DATA DECISIONS
20 Brace Road
Cherry Hill, NJ 08034

20 Brace Road • Cherry Hill, NJ 08034

☐ YES! Please tell me how I can examine **PC COMMUNICATIONS** for a no-purchase-obligation trial review.

C6

Name ____________
Title ____________ Phone (____)________
Company ____________
Address ____________
City, State, Zip ____________

A204

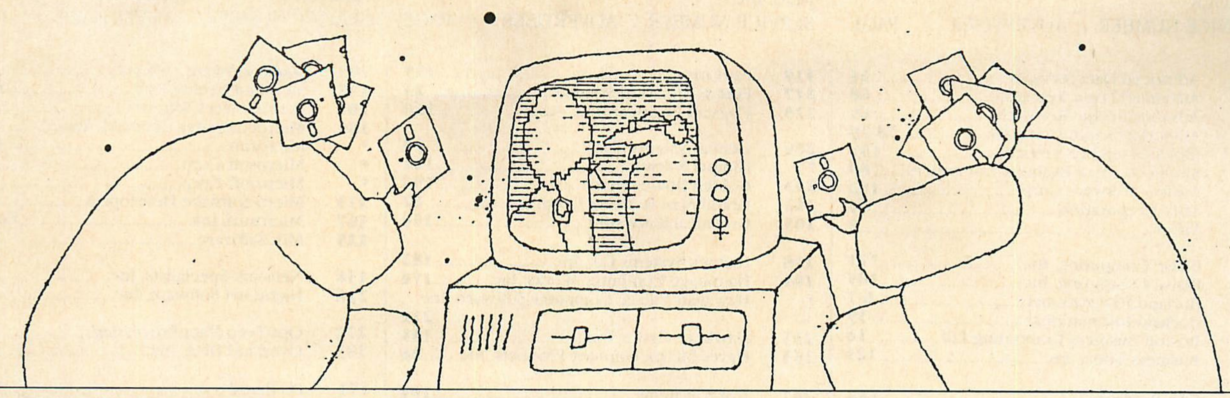


ILLUSTRATION • DAVID POVILAITIS

DECEMBER

December 1-4
10th ACM Symposium on Operating Systems Principles
Eastsound, WA
Contact: Forest Baskett, Digital Equipment Corporation, 4410 El Camino, Los Altos, CA 94022; 415/949-0777

December 2-3
Application Prototyping: A Key to Successful Systems Development
Chicago, IL
Contact: Belinda Selleck, Marketing Coordinator, Digital Consulting Associates, Inc., 6 Windsor Street, Andover, MA 01810; 617/470-3870

December 2-4
Softfair II, 2nd Conference on Software Development Tools, Techniques, and Alternatives
San Francisco, CA
Contact: Softfair 85, P.O. Box 639, Silver Spring, MD 20901; 202/371-0101

December 2-5
Globecom 85: IEEE Global Telecommunications Conference
New Orleans, LA
Contact: D. P. Dodd, Technical Program Chairman, Bell South Services, P.O. Box C360-4th Floor, Birmingham, AL 35283; 205/321-3723

December 2-5
10th Workshop on Computer Elements
Phoenix, AZ
Contact: IEEE Computer Society, P.O. Box 639, Silver Spring, MD 20901; 202/371-0101

December 2-6
Tutorials for Professional Development
Boston, MA
Contact: Gerry Segal, Education Manager, The Association for Computing Machinery, 11 West 42nd Street, New York, New York 10036; 212/869-7440

December 2-6
Micro 18, Microprogramming Workshop
Pacific Grove, CA
Contact: Micro 18, P.O. Box 639, Silver Spring, MD 20901; 202/371-0101

December 3-5
Real-Time Systems Symposium
San Diego, CA
Contact: Andre M. van Tilborg, Carnegie-Mellon University, Pittsburgh, PA 15213; 412/578-3801

December 6-8
Bits & Bytes National Computer Show for Kids
Dallas, TX
Contact: Ed Flynn, Information Processing Group, 87 N. Raymond Avenue, Suite 305, Pasadena, CA 91103; 818/792-5111

December 10-12
1985 Videodisc, Optical Disk, and CD-ROM Conference and Expo
Philadelphia, PA
Contact: Jean-Paul Emard, Meckler Publishing, 11 Ferry Lane West, Westport, CT 06880; 203/226-6967

December 11-13
DEXPO West '85
Anaheim, CA
Contact: Kathy Karas, Expo Council International, 55 Princeton Heightstown Road, Princeton Junction, NJ 08550; 609/799-1661

December 9-13
2nd Conference on Artificial Intelligence Applications
Miami Beach, FL
Contact: Artificial Intelligence Conference, P.O. Box 639, Silver Spring, MD 20901; 202/371-0101

December 11-13
1985 Winter Simulation Conference
San Francisco, CA
Contact: Gerard Blais, FED SIM/CAA, Washington, DC 20330; 202/274-7113

December 16-20
1st International Conference on Supercomputing Systems
Tarpon Springs, FL
Contact: Supercomputing Conference, P.O. Box 639, Silver Spring, MD 20901; 202/371-0101

December 16
Call for papers: 2nd Symposium on Computational Geometry
Yorktown Heights, NY
(June 2-4, 1986)
Sponsor: ACM SIGGRAPH and SIGACT
Contact: David P. Dobkin, Department of Computer Science, Princeton University, Princeton, New Jersey 08544

December 16
Distributed Database Architectures
Washington, DC
Contact: Software Institute of America, 8 Windsor Street, Andover, MA 01810; 617/470-3880

JANUARY

January 8-10
HICSS-19: Hawaii International Conference on System Sciences
Honolulu, HI
Sponsors: University of Hawaii, University of Southwestern Louisiana, and IEEE-CS
Contact: Bruce Shriver, IBM T. J. Watson Research Center, P.O. Box 218, Route 134, Yorktown Heights, NY 10598

January 13-15
13th Annual ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages
St. Petersburg, FL
Sponsor: Association for Computing Machinery
Contact: Edmund Gallizzi, Eckerd College, St. Petersburg, FL 33733

January 13-16
Fifth Symposium on Reliability in Distributed Software and Database Systems
Los Angeles, CA
Contact: IEEE Computer Society, P.O. Box 639, Silver Spring, MD 20901; 301/589-8142

INDEX TO ADVERTISERS

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION
(Required by 39 U.S.C. 3685)

1. a. Title of Publication: PC Tech Journal
   b. Publication No. 07380194.
2. Date of filing: October 1, 1985
3. Frequency of issue: Monthly
   a. No. of issues published annually: 12
   b. Annual subscription price: $29.97.
4. Location of known office of publication: One Park Avenue, New York, New York 10016.
5. Location of the headquarters or general business offices of the publisher: One Park Avenue, New York, New York 10016.
6. Names and complete addresses of the publisher, editor, and managing editor: Publisher, Newton Barrett, One Park Avenue, New York, New York 10016; Editor: Will Fastie, The World Trade Center, Suite 211, Baltimore, MD 21202; Managing Editor: Marjory Spraycar, The World Trade Center, Suite 211, Baltimore, Maryland 21202.
7. Owner: Ziff-Davis Publishing Company, One Park Avenue, New York, N.Y. 10016; Ziff Corporation, One Park Avenue, New York, N.Y. 10016.
8. Known bondholders, mortgagees, and other security holders owning or holding 1 percent or more of total amount of bonds, mortgages or other securities: None.
10. Extent and Nature of Circulation:

	Average No. Copies Each Issue During Preceding 12 Months	Actual No. Copies of Single Issue Published Nearest to Filing Date
A. Total no. copies *(Net Press Run)*	114,166	112,942
B. Paid circulation		
1. Sales through dealers and carriers, street vendors and counter sales	10,700	97,000
2. Mail subscriptions	81,500	85,600
C. Total paid circulation *(Sum of 10B1 and 10B2)*	92,200	95,300
D. Free distribution by mail, carrier or other means Samples, complimentary, and other free copies	1,875	2,424
E. Total distribution *(Sum of C and D)*	94,075	97,724
F. Copies not distributed		
1. Office use, left over, unaccounted, spoiled after printing	1,558	4,867
2. Returns from news agents	18,533	10,351
G. Total *(Sum of E, F1 and 2 —should equal net press run shown in A)*	114,166	112,942

11. I certify that the statements made by me above are correct and complete.

WILLIAM L. PHILLIPS,
Vice President

NEW !!

TURBO
INTEGRATED DEVELOPMENT SYSTEM

NEW !!

These systems were developed from the ground up to work in an integrated fashion ... This saves you time and money from trying to assemble different packages and to code in our Products Unique Features.

TURBO ISAM MASTER

GENERATES 'Ready to Run' Turbo Programs Using BORLAND'S Turbo Pascals Database TOOLBOX.

With a few key strokes you can generate the following Pascal programs/includes files. Automatically interfaces to Turbo Screen Master.

A. MASTER DATABASE PROGRAM - Generates Pascal Program Code for the following functions.

1. Add a Record • Allows both duplicate & unique only keys. • Change the up to 6 fields comprising the primary key and the up to 3 secondary keys at any time during input, and the validity of the keys are checked and the ISAM key files are automatically adjusted.

2. Delete a Record - Shows the record to be deleted on the screen and allows you to change your mind about deleting, and then adjusts the keys files automatically for the up to 4 keys. The disk space is reused automatically by the programs.

3. Edit a Record - with a key change allowed.

4. Search Database by Key - Up to 4 keys-and has a 1 key lookup to find next key in alphabetic order.

5. For Each Key - The ability to display the keys on the screen or printer in sorted order.

B. DATA BASE RECOVERY PROGRAM. This program recovers your data base if it is corrupted by a power outage or certain hardware failures.

C. MULTIPLE ISAM FILES - Can be used at the same time in a single program. The generator also produces context sensitive instructions on how to integrate the generated program into your main program.

D. DOCUMENTATION - Print screens and ISAM specifications. Also inline program documentation is generated.

TURBO MENU MASTER

GENERATES 'Ready to Run' Turbo Programs with Dedicated Screens or Windows.

Features: Has Menu Data Base so its easy to modify menues.

Provides Selection by - **(1) Press of a number**-If you have more than 9 selections then you enter only a single number if the selection is 2-9. **(2) Press of a function key** - if more than 10 selections then can enter use shift function key to get up to 18 selections. **(3) Press the High Lighted Letter** -in the selection description, you choose a letter for each menu selection in the interactive menu generator. **(4) Use the arrow keys** - to point to a menu item & then press return.

Adaptive Screen Colors -Different screen colors are automatically used for color or monochrome display. This allows you to provide a better interface for the user.

Interactive Menu Builder - (1) Allows for the automatic entry & reorder of selections. (2) Offers easy color selection. (3) Allows for choice of procedure, chain, execute or comment code generation for each selection. (4) Has startup Menu that YOU design. (5) Provides for the integrated Display & Control of the Key Lock Status.

RISK FREE TRIAL

Try the demo package included for 30 days. If not pleased return for a full refund.

NO ROYALTIES on Generated Programs

Credit card orders call:

1-800-821-9503

In Florida (800) 342-0137

HAWAIIAN VILLAGE COMPUTER SOFTWARE
1109 Pennsylvania Ave. St. Cloud, Florida 32769

TURBO SCREEN MASTER

Some of the **UNIQUE** features of the **Turbo Screen Generator** are:

Intelligent Cursor - Computer automatically determines which single/double character to use from the Extended IBM Character set to draw boxes and borders. You just move the cursor where you want the border. The other Screen Systems require that you must tediously choose the correct graphic character.

Extended Variable Names - Allows the use of Structures & Arrays in your variable names and automatically generates the correct declarations and initalization routines. Also you are allowed to declare a field larger than the screen display size.

Provides Three Color Selections Methods - Easy to develop color screen on Monochrome Monitor and to develop Monochrome Screens on the Color Monitor.

Adaptive Screen - Entry colors change automatically depending on what type of screen is in use.

COMPARISON WITH OTHER PRODUCTS	TURBO SCREEN MASTER	TURBO SCREEN VER - 1.10	SCREEN SCULPTURE Ver - 1.01
Border Color Control	YES	NO	NO
You Assign Variable Names	YES	NO	YES
No cyptic variable names whose names depend on where they appear on screen			
Range and Date Checks	YES	NO	YES
Data Entry Valid Character Set	YES	NO	NO
Data Entry Mask	YES	NO	YES
Helpful for Profession Screen Input & Validation			
Initalize Variables to a starting value	YES	NO	YES
Data Entry Valid String Set	YES	NO	NO
Pascal storage for type of Boolean & Integer	YES	YES	NO
Control Capa/Num Look	YES	NO	NO
Auto-Initalization of Date/Time	YES	NO	NO
User Defined Error & Message Handler	YES	NO	NO
Generated program adapts automatically to IBM Screen-Monitor Type	YES	NO	YES
Handles Function Keys	YES	NO	NO
Help Screen Procedures	YES	YES	NO
Optional ISAM Keys Screens Code Generated automatically	YES	NO	NO
Turbo Toolkit Included	YES	NO	NO
Undo Function	YES	NO	NO
Provides Running Time Display	YES	NO	NO
Integrated Data Dictionary	YES	NO	NO
Auto-start Screen for Screen Generation	YES	NO	NO
PRICE (INCLUDES SHIPPING)	$89.95	54.95	125.00

FREE TURBO TOOL KIT $44.95 VALUE

WITH ORDER OF A MASTER PRODUCT

YES, Enclosed is ______________

INTRODUCTORY OFFER

☐ Turbo Pascal 3.0	~~$69.95~~	$59.95
☐ Turbo Database Tool Box	~~$54.95~~	$49.95
☐ Turbo Menu Master	~~$69.95~~	$54.95
☐ Turbo Screen Master	~~$124.95~~	$89.95
☐ Turbo ISAM Master	~~$149.95~~	$124.95

Systems Requires: IBM PC,XT,AT or 100% Compatiable -196K. MS DOS 2.0 or Higher - 80 column screen. Turbo Pascal 3.0 - 2 DD/DS Disk Drives. ISAM Program Generator Requires Turbo Database Toolbox.

*These prices include shipping to all U.S. Cities. All foreign orders add $10 per product ordered

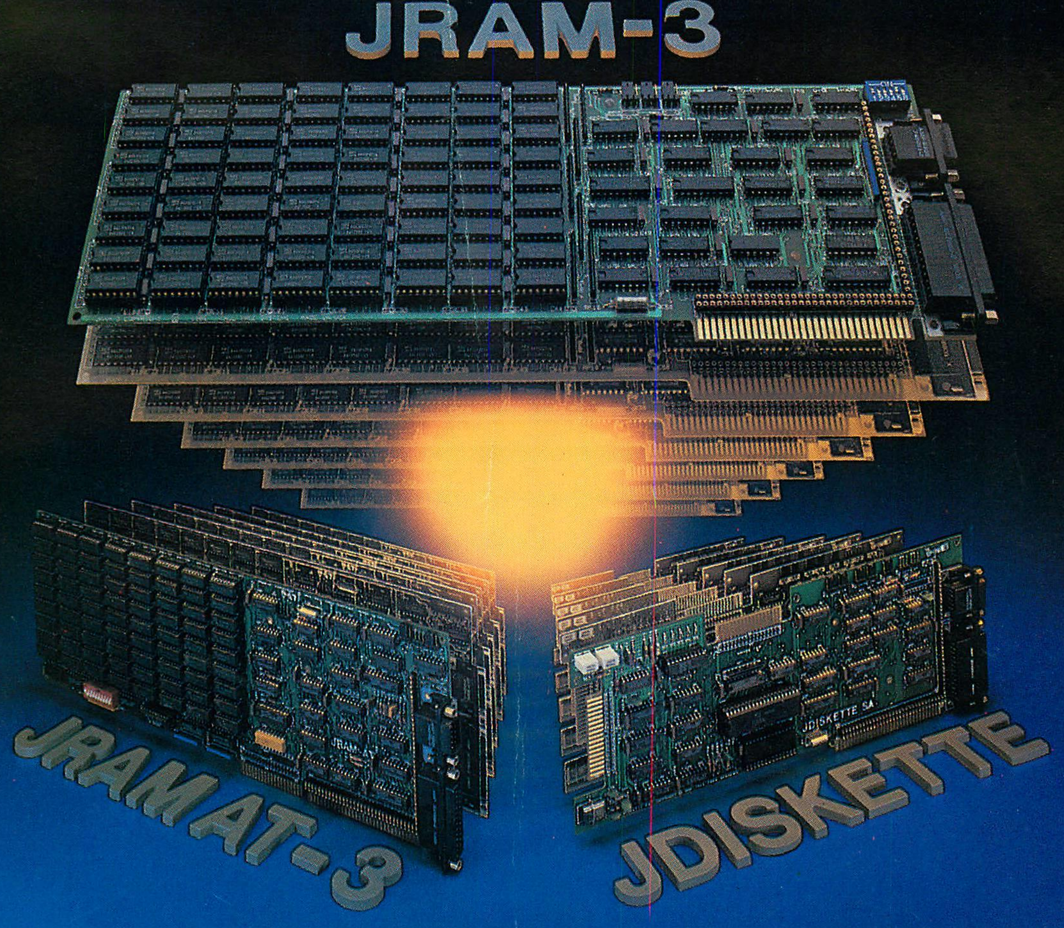
For Other Inquires Call (305) 892-5686

Name: ______________ **Phone:** ______________
Shipping Address: ______________
City: ______________ **State:** ______________ **Zip:** ______________
VISA or MC #: ______________ **Exp. Date:** ______________

C.O.D.'s and Purchase orders WILL NOT be accepted. Outside USA: make payment by bank draft, payable in US dollars drawn on a US bank.

INDEX TO PRODUCTS

THE FAMILY TREE



BRANCHING OUT

Tall Tree Systems presents JRAM-3, the newest member of the JRAM family. JRAM-3 is a fourth generation multifunction memory board and the successor of the highly praised JRAM-2. Designed to meet the latest expanded memory specification standard being implemented by the major spreadsheet vendors, JRAM-3 can access up to eight megabytes of memory for larger, more efficient spreadsheets. JRAM-3 can also be used for DOS memory, electronic disk, print spooler, and program swapping applications!

Determined to maintain our reputation as the price leader in memory expansion, Tall Tree Systems offers JRAM-3 fully populated with two megabytes for an amazing $699. A JETDRIVE/JSPOOL combo disk is included free of charge. This is the same highly acclaimed JRAM software that has helped make Tall Tree Systems the pioneer in the industry for bank switched memory and RAM disk technology. The new combo disk features JPAGER which allows any expanded memory application program to utilize multi-megabyte memory.

The family tree wouldn't be complete without an AT version. Look to us in September for the new JRAM AT-3, our third generation 16 bit board providing support for expanded spreadsheets. JRAM AT-3 will be available with two megabytes at $799.

Don't forget the rest of the JRAM family either. We'll continue producing our popular JRAM-2 and our present JRAM AT board, along with our full range of "slim daughter board" modules. The JRAM-2 and JRAM-3 can co-exist in the same computer, but only the JRAM-3 can be used for expanded memory applications. Use JRAM-2 for DOS, multi-tasking, print spooling, and electronic disk.

To complete the family line-up we feature JDISKETTE. This controller card lets you put 1.2 megabyte diskette drive(s) in your PC or XT. You can read, write, and format the AT high density diskettes and communicate easily between a PC or XT and an AT. You won't lose a single slot, since JDISKETTE replaces your old controller and handles up to four internal floppy drives. Free of charge with JDISKETTE, you will receive the popular JFORMAT software, a DOS compatible device driver that gives your computer the ability to handle expanded disk formats.

Welcome to the family!

CIRCLE NO. 197 ON READER SERVICE CARD

TALL TREE SYSTEMS
1120 San Antonio Road
Palo Alto, CA 94303
(415) 964-1980

The Transporter Night Shift

Long After Your Staff Goes Home, We Keep Your PC Making Calls, Sending and Receiving Files, and Keeping a Log

Your microcomputer doesn't go home at five. Why not keep it working?

TRANSPORTER keeps your PC communicating almost as well as your staff could do it. Just load it with a "task list" before you go home. It can automatically call other computers, log in, repeat calls if necessary, send and receive files, and more.

It can do it overnight, when phone rates are lowest.

The next morning, TRANSPORTER gives you a report of what it has accomplished during its night shift.

So you can start the day ahead of the game.

TRANSPORTER comes with our popular CROSSTALK data communications program added on. It communicates with any other system running TRANSPORTER or CROSSTALK. For details, see your retailer or write for a brochure.

MICROSTUF®
1000 Holcomb Woods Parkway
Roswell, Georgia 30076

For the IBM Personal Computer. Requires 128K RAM, two disk drives, PC-DOS, Asynchronous Communications interface or equivalent RS232 connection, and a modem compatible with the 'AT' command set.

TRANSPORTER™

CIRCLE 167 ON READER SERVICE CARD

TECH
JOURNAL
P.O. Box 2966
Boulder, Colorado 80321